FIFTH SEMI-ANNUAL REPORT

ON

RESEARCH ON

# CONTROL OF FREE-FLYING

# SPACE ROBOT MANIPULATOR SYSTEMS



Submitted to

Dr. Henry Lum, Jr., Chief, Information Sciences Division
Ames Research Center, Moffett Field, CA 94035

by

The Stanford University Aerospace Robotics Laboratory
Department of Aeronautics and Astronautics
Stanford University, Stanford, CA 94305

Research Performed Under NASA Contract NCC 2-333
During the period March 1987 through August 1987

Professor Robert H. Cannon Jr.
Principal Investigator

FIFTH SEMI-ANNUAL REPORT

ON

RESEARCH ON

# CONTROL OF FREE-FLYING SPACE ROBOT MANIPULATOR SYSTEMS

Submitted to

Dr. Henry Lum, Jr., Chief, Information Sciences Division
Ames Research Center, Moffett Field, CA 94035

by

The Stanford University Aerospace Robotics Laboratory
Department of Aeronautics and Astronautics
Stanford University, Stanford, CA 94305

Research Performed Under NASA Contract NCC 2-333
During the period March 1987 through August 1987

Professor Robert H. Cannon Jr.
Principal Investigator

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

This document reports on the work done under NASA Cooperative Agreement NCC 2-333 during the period March 1987 through August 1987. The research is carried out by a team of graduate students comprising the Stanford University Aerospace Robotics Laboratory under the direction of Professor Robert H. Cannon, Jr. The goal of this research is to develop and test new control techniques for self-contained, autonomous free-flying space robots. Free-flying space robots are envisioned as a key element of any successful long term presence in space. These robots must be capable of performing the assembly, maintenance, and inspection, and repair tasks that currently require astronaut extra-vehicular activity (EVA). Use of robots will provide economic savings as well as improved astronaut safety by reducing and in many cased eliminating the need for human EVA.

The focus of our work is to develop and carry out a set of research projects using laboratory models of satellite robots. These devices use air cushion technology to simulate in two dimensions the drag-free, zero-g conditions of space. Using two large granite surface plates (6' by 12' and 9' by 12') which serve as the platforms for these experiments we are able to reduce gravity induced accelerations to under $10^{-5}g$ with a corresponding drag-to-weight ratio of about $10^{-4}$—a very good approximation to the actual conditions in space.

Our current work is divided into four major projects or research areas: Cooperative Manipulation on a Fixed Base, Cooperative Manipulation on a Free-Floating Base, Global Navigation and Control of a Free-Floating Robot, and an alternative transport mode called LEAP (Locomotion Enhancement via Arm Push-Off).

The fixed-base cooperative manipulation work represents our initial entry into multiple arm cooperation and high-level control with a sophisticated user interface. This experiment is just coming on-line and should be fully operational shortly.

The floating-base cooperative manipulation project strives to migrate some of the new technologies developed in the fixed-base work onto a floating base. This experiment will be using our second generation space-robot model which is still under construction.

The global control and navigation experiment seeks to demonstrate simultaneous control of the robot manipulators and the robot base position so that tasks can be accomplished while the base is undergoing a controlled motion.

The LEAP project is a new activity that was started during this report period with

the goal of providing a viable alternative to expendable gas thrusters for vehicle propulsion wherein the robot uses its manipulators to throw itself from place to place. This work will be carried out with a slightly revised version of the second generation space robot which is currently under construction.

The chapters that follow give detailed progress and status reports on a project by project basis. Also undergoing final preparation is Harold L. Alexander's PhD thesis entitled "Experiments in Control of Satellite Manipulators." This document will provide an in-depth report on the initial satellite robotics work done at Stanford University.

# Chapter 2

# Fixed-Base Cooperative Manipulation Experiment

Stan Schneider

## 2.1 Introduction

To accelerate our development of multi-armed, free-flying satellite manipulators, we are developing a fixed-base cooperative manipulation facility. Although the manipulator arms are fixed, they will manipulate free-flying objects. By allowing allow us to quickly experiment with cooperative algorithms, this facility will greatly expedite our study of space-based manipulation and assembly. This section describes the progress made to date in our research on cooperative manipulation.

### Progress Summary

The major activities completed during during the period March, 1987 through August 1987 were:

- Constructed the cooperating-arms experimental system
- Developed multiprocessor real-time computer system
- Automated the inertia measurement device, and measured arm link parameters
- Began development of the real-time software
- Demonstrated simple single-arm control

### Background: research goals

Space construction requires the manipulation of large, delicate objects. Single manipulator arms are incapable of quickly maneuvering these objects without exerting large local

torques. Multiple cooperating arms do not suffer from this limitation. Unfortunately, cooperative robotic manipulation technology is not yet well understood. The goal of this project is to study the problem of cooperative manipulation in a weightless environment, and experimentally demonstrate a cooperative robotic assembly.

Four aspects are to be studied in detail:

- The dynamic control of multiple arm manipulation systems
- The utilization of video "vision" data for real-time control
- Real-time software structuring for cooperative robotic systems
- User interfacing: the acquisition and utilization of strategic commands

## 2.2 Facility Development

During this period, the mechanical hardware and computer system design described in the previous report was realized. The fixed-base cooperation facility consists of a pair of two-link manipulators, affixed to the side of a "small" granite table (4 x 8 feet). Each arm is of the popular SCARA configuration—basically anthropomorphic, with vertical-axis, revolute "shoulder" and "elbow" joints. The arms are capable of motion in the plane of the table, and can interact with air-cushion objects floating on the granite surface.

The arms were designed with two major goals: to be compatible with the SRSV design, and to be as simple as possible. The SRSV design constraints are described elsewhere.

### Manipulator design

Each link is directly driven by a limited angle torque motor. These motors were chosen for their nearly frictionless operation. The elbow joint is driven remotely via a cable and pulley system. This allows the elbow motor to be located at the shoulder base, and thus drastically reduces the effective inertia of the upper-arm link. It also permits a 3:2 gear reduction in the elbow drive train, extending the range of motion of the elbow joint.

### Joint angle sensors

Each joint is equipped with a rotary variable differential transformer (RVDT). These sensors provide direct angular position measurements. To provide an estimate of the joint angular velocity, the position signals are passed through a pseudo-differentiation filter. This filter has the transfer function $\frac{s}{(s+a)(s+b)}$. Thus, at low frequencies (small s), it approximates $\frac{s}{ab}$, a differentiation operator. Unfortunately, this filter does introduce some phase lag, but it is not significant at the low (almost D.C.) frequencies of importance to this rigid body system.

### Force sensing gripper

In order to be able to manipulate targets in a cooperative manner we have developed a force-sensing end effector. A pneumatic actuator drives a beam assembly in the vertical direction providing positive attach and release functions. Strain gauges on the beam provide force signals in the two planar directions. In preliminary calibration tests, the device was capable of a very high signal-to-noise ratio—on the order of 2000:1. Crosstalk and non-linearities in the strain gauges are less ideal.

### Vision system

Completion of the tasks required of our robots requires sensing not only the robot's motions, but also those of objects in its environment. To allow this, we are developing a simple computer vision system. This system is capable of tracking specially designed variable gray-scale "targets". Considerable theoretical work was done (under a previous contract) to analyze the sub-pixel tracking performance of a vision system employing these targets. An excerpt of that work is attached as appendix A.

### Floating object development

Our floating air-cushion objects have also undergone evolutionary development since the last report. These objects are independent miniature air-cushion vehicles, equipped with a small battery powered aquarium-pump air supply. The arms can manipulate them with their grippers, thus providing a two-dimensional simulation of space-based manipulation.

We have developed several prototype models for these objects. The original design had oval-shaped pads with several air outlet holes and flow restriction orifices. While this design worked under most conditions, we found that the small aquarium pumps developed insufficient air flow through the orifices. After several iterations, we have settled on a design with only two outlet holes, but with larger air plenums on the lower surface of the pad. Dual pumps provide each plenum with an independent air supply. This design permits relatively large off-center loading, while still permitting fast object motions on the limited air pressure developed by the portable pump.

## 2.3 Computer System

Our real-time computer system combines a proven UNIX development environment with high performance real-time processing hardware. Motorola 68020/68881 single-board processors running the real-time kernel pSOS provide inexpensive real-time processing power. VME bus shared-memory communications permit efficient multiprocessor operation. The real-time processors are linked, via the VME bus, to our Sun/3 engineering workstations. Thus, we benefit from Sun's superb programming environment, while providing the capacity for relatively cheap, unlimited processing expansion.

### Real-time hardware

The fixed-base facility computer system was configured and installed during the report period. This system employs three Motorola MVME-133 processor cards. These cards each feature a 68020 CPU, a 68881 floating point co-processor, and 1 Mbyte of memory. The system communicates with our Sun workstation via a VME bus repeater.

Both analog and digital interfaces are provided to the robotic hardware. A/D converters process the incoming RVDT and force sensor signals. The motors are driven by a simple D/A converter and a current amplifier. The pneumatic gripper actuator is interfaced to the system via a digital interface card.

### Real-time software

We are actively developing a software link between our Sun workstations and the real-time computer system. We have successfully utilized the Sun's native "C" compiler to produce and down-load real-time system code. A more general and powerful real-time programming environment is under development.

## 2.4 Calibration

The lab's inertia measurement device was automated during this report period, and all arm inertia and mass parameters were measured. This section describes the calibration methodology.

The ARL's inertia measuring device is a simple 3-wire rotary pendulum, with a plate to hold the measured part. The pendulum's period is related to the inertia of the part on the plate. The pendulum and its dynamics have been described in previous reports.

In the past, the pendulum period had been measured manually, and rather tedious calculations were required to calculate the unknown part's inertia. By using an LED and photodiode to sense the motion of the swinging pendulum, the real-time system can calculate the pendulum's period. The manual process has been replaced by a simple automated sequence: After the device is calibrated with two know inertias, unknown inertias may be quickly measured. All calculations are done automatically by the inertia measurement program.

## 2.5 Controllers

We are examining interfaces between the dynamic forces and motions of the robotic manipulators, and higher-level strategic inputs. As a first attempt, we are investigating the application of Nevill Hogan's[2] impedance control concept to multi-arm—and multi-vehicle—cooperative tasks. Impedance control is very attractive for cooperative tasks because it

allows direct control of the interaction between the cooperating agents; control of the mechanical power flow from manipulation system to environment. The implementation for multiple arms, however, is not well understood.

We were successful in implementing a simple Position-Derivative (PD) controller for a single arm during this report's period. This accomplishment demonstrated the viability of all the sensor and actuator sub-systems. It also underlined the need for extensive calibration and computer software efforts.

## 2.6 Future Work

During the next period, work will continue on the construction and calibration of the cooperating arms experimental system. This should allow development of much more sophisticated dynamic control algorithms. We will also continue our software development. The second arm should be operational soon, and we expect to implement our first dual-arm controller during the next report period.

~

# Chapter 3

# Multiple Arm Cooperation on a Free-Flying Robot

Ross Koningstein

## 3.1 Introduction

This chapter summarizes the work performed on multiple arm cooperation on a free-flying robot. This work represents one of the basic technologies required for space based manipulation. One of the first steps to achieving control of a system is to understand its dynamic properties. This section and an included appendix discuss the work done in the dynamic modelling of a typical space robot configuration: the kinematic chain.

### 3.1.1 Motivation

To achieve fast, precise control of a physical system, accurate dynamical modelling is required. As controlled dynamical systems become increasingly complex, manual derivation by an analyst of the system equations of motion becomes very costly, and human error limits the rate at which a dynamical model can be created. Computer codes for generating dynamical equations of motion have appeared, however, these codes have neglected the needs of the control system designer since they approach the problem purely from a simulation viewpoint. Automatic equation generation packages are available, however, unfortunately, in order to control a dynamical system, the system's Jacobian, **J** as defined by

$$\mathbf{v}^{\text{endpoints}} = \mathbf{J}\dot{q}$$

where **v** is a vector of the speeds of the manipulator endpoints, measured in some coordinate system and $\dot{q}$ are the derivatives of the system generalized coordinates. The Jacobian and its derivative need to be used for computation of the desired controls. We will demonstrate that the Jacobian can be computed from the same terms used to compute the dynamical equations.

A secondary motivation is that the dynamical system under study, a dual arm satellite manipulator model, is essentially a serial chain of rigid bodies and undergoes only minor changes (in terms of structure) when it grasps an object: it either becomes a longer chain or it becomes a closed chain. If the equations of motion of a chain system have a certain form, then the addition of extra links to the system should result in a small change in the computation of the equations of motion and should not necessitate the rederivation of the system's equations of motion from scratch. Generating equations of motion and control algorithmically is desirable since this task is then no longer a manual procedure. It will require significantly less of the analyst's time and will be less susceptable to error.

## 3.2 Dynamical Modelling of Free-Flying Kinematic Chains

The space robot being considered falls into the class of objects called kinematic chains. The mathematical model for kinematic chains has a special structure allowing an algorithm to formulate dynamical equations for arbitrarily long chains. This algorithm can be highly efficient, and much of the work done for the dynamic modelling can also be used for control specifications and dynamic constraint equations if the chain becomes closed. The theory for serial chain manipulators is derived using Kane's dynamical analysis techniques[3]. The theory showing the formulation of equations of motion, the system Jacobian, the control specifications and closed loop constraints is outlined in appendix B. Using the algorithm developed, it is possible to simulate kinematic chains of arbitrary size by specifying the masses and inertias of the bodies in the chain as well as their interconnections. No further work by the analyst is necessary. Note that the algorithm developed to date is not for general dynamical systems: it does not handle closed chains nor does it handle three dimensional systems. It does, however, provide us with a very powerful tool for the study of dynamical chain systems, an example of which is the satellite simulator robot on which we wish to test control algorithms for free-flying robots.

Simulation runs of a dual armed two-dimensional free-flying robot were performed. The correctness of the algorithm for dynamical modelling was confirmed both by checking the conservation of the system's Hamiltonian and by comparison to runs of similar equations produced by SDEXACT[6].

## 3.3 Status

Equations of motion for a dual-armed free-flying satellite robot simulator have been derived manually and numerically formulated using an algorithm which computes terms of the equations of motion. The addition or removal of links of the robot (e.g. to simulate system configuration before and after object catches) requires changing only the kinematic chain description data structure.

The algorithm developed has been used to generate numerical dynamical equations for the free-flying two armed robot. These equations were solved, and the resulting history of coordinates and speeds provided an accurate simulation of the robot's dynamics. To date, only free (uncontrolled) motion of the satellite robot model has been simulated.

## 3.4 Further Research

The speed of the numerical solution to the numerical dynamical equations can be improved for large systems (e.g. three dimensional multi-armed robots) by using a different type of numerical dynamical equation. The algorithm derived is an order $n^3$ algorithm for simulation and control. We also wish to investigate the newer the order $n$ simulation algorithms being developed[5]. The mastering of this theory will allow us to approach the control problem of the free-flying robot with fast and accurate dynamics and control specification model. Our long run goal is to test this theory on a two-armed two-dimensional satellite robot manipulator model in order to investigate the limitations of such theory in real-world situations.

# Chapter 4

# Navigation and Control of Free-Flying Space Robots

Marc Ullman

## 4.1 Introduction

This chapter summarizes the progress to date in our research on global navigation and control of free-flying space robots. This work represents one of the key aspects of our comprehensive approach to developing new technology for space automation. Ultimately, we envision groups of fully-self contained mobile robots serving as the core work force in space.

### 4.1.1 Motivation

Although space presents us with an exciting new frontier for science and manufacturing, it has proven to be a costly and dangerous place for people. Space is therefore an ideal environment for sophisticated robots capable of performing tasks that currently require the active participation of astronauts.

### 4.1.2 Research Goals

The immediate goals of this project are to:

- demonstrate the ability to simultaneously control vehicle position and arm orientation so that a robot can navigate to a specified location in space while manipulating its arm(s).
- demonstrate the ability to capture a (possibly moving) free-floating target "on-the-fly" using the manipulator arm while the base is in transit.
- provide a suitable platform for the eventual addition of A.I. based path planning and obstacle avoidance algorithms which will enhance the robustness of task execution.

### 4.1.3 Background

This work emphasizes the modeling of robot dynamics and the development of new control strategies for dealing with problems of:

- a non-inertially fixed base (i.e. free-floating base)
- redundancy with dissimilar actuators
- combined linear and non-linear actuators
- highly non-linear dynamics
- unstructured environments

Our laboratory work involves the use of a model satellite robot which operates in two-dimensions using air-cushion technology. We have developed a series of satellite robots which, in two dimensions, experience the drag-free and zero-g characteristics of space. These robots are fully self-contained vehicles with onboard gas supplies, propulsion, electrical power, computers, and vision systems. The latest generation of robots is also equipped with a pair of two-link arms for acquiring and manipulating target objects.

## 4.2 Summary of Progress

We have made important strides during the past six month period on both the theoretical and experimental fronts. In the area of hardware development, the new robot vehicle whose design was described in our last report has continued to materialize. We have completed construction and demonstrated operation of the onboard thrust subsystem. We have also had a pair of the manipulator arms fabricated and instrumented. These arms are now mounted on the robot. A successful closed loop controller has been demonstrated in simulation which simultaneously controls the vehicle position (using on-off-on thrusters) and the arm orientation (using torque motors).

## 4.3 Experimental Hardware

The current generation robot has been designed to accommodate one or two two-link rigid arms mounted between the first and second layers as described in our previous report. These arms have been designed and fabricated and now await final wiring of sensors and motors. These arms are identical to those used in the fixed-base cooperating-arms experiment and are described in greater depth in Chapter 2. This hardware commonality will greatly simplify the transfer of technology from fixed based to free-floating robots.

## 4.4 Modeling and Simulation

The complete dynamical equations of motion have been derived and verified for a single-armed version of the robot. These equations haved been coded up and simulated for both free and forced motion.

### 4.4.1 Analytical Model

The robot has initially been modeled with only one arm since the global control and target capturing problems can be addressed with this somewhat simpler configuration. (See the section on Multi-Arm Cooperation for a derivation of the equations of motion for the two-armed version.) The model consists of three planar rigid bodies connected by two torque motors. (See Figure 4.1). The base body is capable of translation and rotation in the plane via eight on-off-on thrusters mounted as 90° opposed pairs on each of four corners.

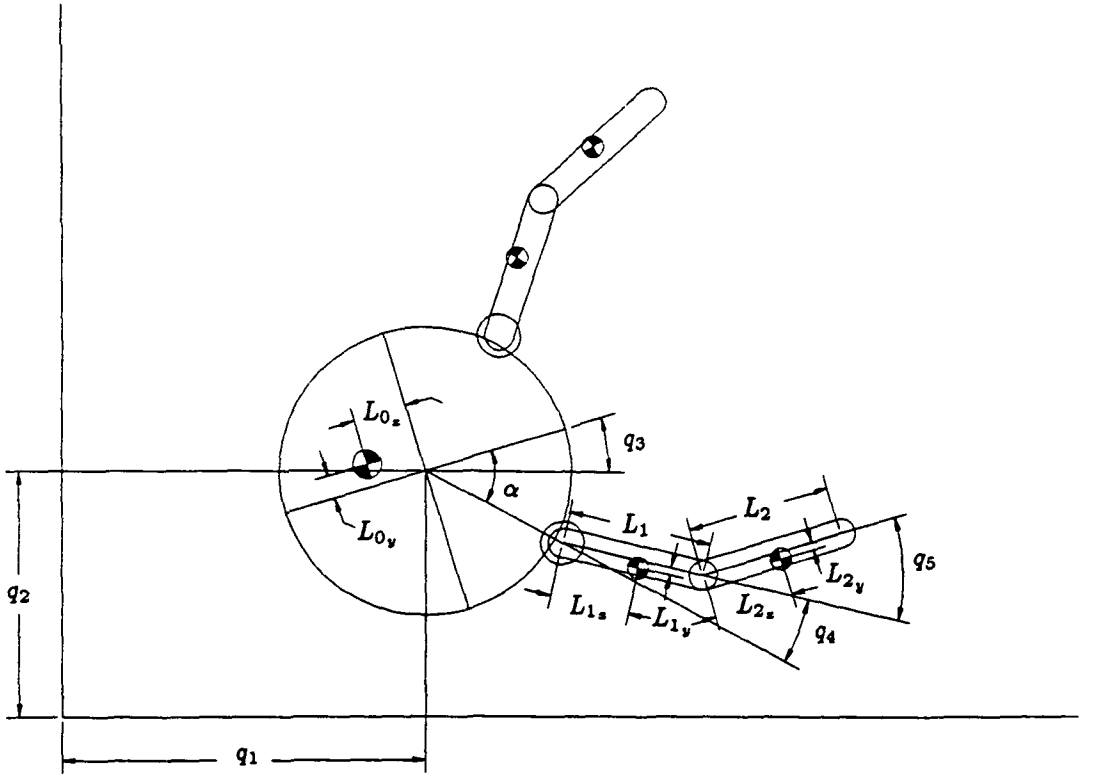



Figure 4.1: Free body diagram of space robot indicating nomenclature used for dynamic modelling.

### 4.4.2 Equations of Motion

The equations of motion for this five-degree-of-freedom system were derived using Kane's method[3] and for verification purposes were also derived using the symbolic equation

generation program SDEXACT[6]. The equations are in the form

$$F_r + F_r^* = 0$$

or

$$\mathbf{F} - M(\mathbf{q})\dot{\mathbf{u}} - V(\mathbf{q},\mathbf{u})\mathbf{u} = 0$$

where

$$\mathbf{F}^* = -(M(\mathbf{q})\dot{\mathbf{u}} + V(\mathbf{q},\mathbf{u})\mathbf{u})$$

$M(\mathbf{q})$ is the configuration dependent mass matrix, $V(\mathbf{q},\mathbf{u})$ is the configuration and velocity dependent matrix of non-linear terms, and $\mathbf{F}$ is the vector of generalized active forces. The $\mathbf{u}$'s or generalized speeds are defined in terms of the state derivatives, $\dot{\mathbf{q}}$, by the relation

$$\mathbf{u} = Y\dot{\mathbf{q}}$$

where

$$Y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

In order to implement a simulation, we must solve the previous set of equations to obtain

$$\dot{\mathbf{q}} = Y^{-1}\mathbf{u}$$

$$\dot{\mathbf{u}} = M(\mathbf{q})^{-1}(\mathbf{F} - V(\mathbf{q},\mathbf{u})\mathbf{u})$$

The vector of generalized active forces is composed of the net forces and torques applied to system and is given by

$$\mathbf{F} = C_{RF}\mathbf{R}$$

where

$$C_{RF} = \begin{bmatrix} -c3 & s3 & s3 & c3 & c3 & -s3 & -s3 & -c3 & 0 & 0 \\ -s3 & -c3 & -c3 & s3 & s3 & c3 & c3 & -s3 & 0 & 0 \\ r_j & -r_j & r_j & -r_j & r_j & -r_j & r_j & -r_j & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

and

$$\mathbf{R} = \begin{bmatrix} R_1 & R_2 & R_3 & R_4 & R_5 & R_6 & R_7 & R_8 & T_1 & T_2 \end{bmatrix}^T$$

The $R$'s are the forces produced by the eight thrusters while the $T$'s are the torques produced by the shoulder and elbow motors respectively.

A listing of the assumed mass and length parameters is given in Table 4.1.

Table 4.1: Assumed Mass and Inertia Properties Used in Simulation

| | | | |
|---|---|---|---|
| Mass of Base | $M_B$ | 110 lbm | 50 kg |
| Mass of Upper Arm | $M_{L1}$ | 2.2 lbm | 1 kg |
| Mass of Fore Arm | $M_{L2}$ | 2.2 lbm | 1 kg |
| Radius of Base | $R_B$ | 9.5 in | .24 m |
| X-axis Base CM Offset | $L_{0_x}$ | 1 in | .025 m |
| Y-axis Base CM Offset | $L_{0_y}$ | 2 in | .051 m |
| Length of Upper Arm | $L_1$ | 12 in | .30 m |
| X-axis Upper Arm CM Offset | $L_{1_x}$ | 6 in | .15 m |
| Y-axis Upper Arm CM Offset | $L_{1_y}$ | 1 in | .025 m |
| Length of Fore Arm | $L_2$ | 12 in | .30 m |
| X-axis Fore Arm CM Offset | $L_{2_x}$ | 6 in | .15 m |
| Y-axis Fore Arm CM Offset | $L_{2_y}$ | 1 in | .025 m |
| Moment of Inertia of Base | $I_{B_{zz}}$ | 5 in | .13 kg m$^2$ |
| Moment of Inertia of Upper Arm | $I_{L1_{zz}}$ | | .13 kg m$^2$ |
| Moment of Inertia of Fore Arm | $I_{L2_{zz}}$ | | .13 kg m$^2$ |
| Thruster Action Line Offset | $r_{jet}$ | 5 in | .13 m |

### 4.4.3 Computer Simulation

The equations of motion derived above have been coded in C for computer simulation. The simulation activity was carried out using the matrix manipulation program PC-Matlab[4]. C and Fortran subroutines can be dynamically invoked from within PC-Matlab allowing for very rapid algorithm prototyping and development. Two integrators were used to check for numerical accuracy, namely a two-evaluation improved Euler method and a four-evaluation fixed-step Runga-Kutta method.

### 4.4.4 Computed Torque Controller

A computed torque[1] controller was implemented as a first cut at closed-loop control. The input commands were all of the form of time and amplitude scaled unit step fifth order "minimum jerk" polynomial trajectories in state (joint) space. Fifth order polynomials were selected so that all trajectories began and ended with zero velocity and acceleration. These are of the form

$$\dot{\mathbf{q}}_d = \mathbf{A}(6\tau^5 - 15\tau^4 + 10\tau^3)$$

$$\dot{\mathbf{q}}_d = \frac{\mathbf{A}}{t_f}(30\tau^4 - 60\tau^3 + 30\tau^2)$$

$$\ddot{\mathbf{q}}_d = \frac{\mathbf{A}}{t_f^2}(120\tau^3 - 180\tau^2 + 60\tau)$$

where $A$ is an amplitude scale factor and $\tau = t/t_f$ is normalized time.

The computed torque controller views the plant as a set of n decoupled second order systems and produces a vector of required accelerations $\mathcal{T}$ given by

$$\mathcal{T} = \ddot{\mathbf{q}}_d + K_v(\dot{\mathbf{q}}_d - \dot{\mathbf{q}}) + K_p(\dot{\mathbf{q}}_d - \mathbf{q})$$

where $K_p$ and $K_v$ are matrices of position and velocity feedback gains. The required acceleration $\mathcal{T}$ is then fed though an inverted model of the plant dynamics (thus giving rise to the alternate name "inverse dynamics") to produce the required generalized active forces (i.e. the computed torques).

$$\mathbf{F} = M(\mathbf{q})\mathcal{T} + V(\mathbf{q}, \mathbf{u})\mathbf{u}$$

where $M(\mathbf{q})$ and $V(\mathbf{q}, \mathbf{u})$ are the mass and nonlinear matrices cited above. Since $\mathbf{F}$ is an $n$ x 1 or in our case a 5 x 1 vector and we have 10 actuators (eight thrusters and two torque motors) we observe that, mathematically speaking, our system has redundant degrees of freedom, i.e. it is underconstrained.

Since each thruster can produce only a positive force (i.e. push rather than push or pull) we can pair them off in four sets which are capable of producing signed forces. (See Figure 4.2) This leaves us with six actuators so we still have one redundant degree of freedom. This redundancy issue is resolved using the classic pseudo-inverse technique. We have

$$C_{RF'} = \begin{bmatrix} -c3 & s3 & c3 & -s3 & 0 & 0 \\ -s3 & -c3 & s3 & c3 & 0 & 0 \\ r_j & r_j & r_j & r_j & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

or $C_{RF}$ with the second, fourth, sixth, and eighth columns removed which leads to

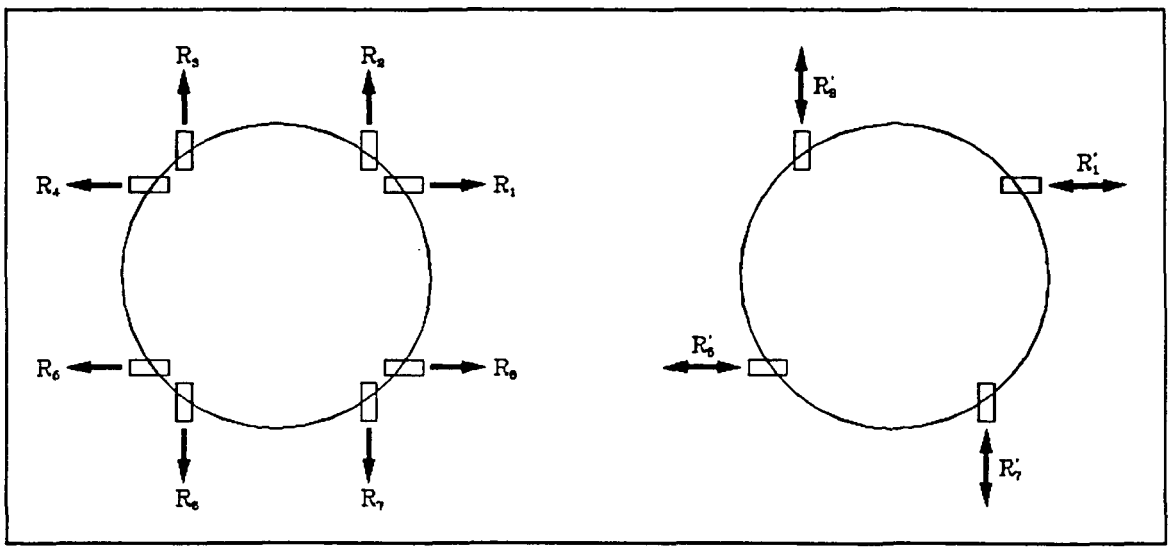
$$\mathbf{R}' = C_{RF'}^{\dagger}\mathbf{F}$$

where † indicates the Moore-Penrose pseudo-inverse[7].

Since the thrusters are on-off devices capable of delivering only one force level, we must pass $\mathbf{R}'$ through a thresholding function as shown in table 4.2. This function also maps the four bi-directional thrusters back into eight uni-directional devices. Figure 4.3 shows a block diagram representation of the complete closed loop simulation with the computed torque controller.

## 4.5 Summary

Progress on robot construction has been moving forward. Now that the robot has arms, the bulk of the mechanical work is complete—the arms still need end effectors and we will probably use the same design as is being used in the fixed-base work described in Chapter 2.




Satellite robot with 8 uni-directional thrusters

Equivalent satellite robot with 4 bi-directional thrusters

Figure 4.2: Modelling uni-directional thrusters with their bi-directional equivalents

Table 4.2: Thruster Thresholding and Mapping Function

| | |
|---|---|
| $R'_1 > R_{th}$ | $R_1$ |
| $R'_7 < -R_{th}$ | $R_2$ |
| $R'_3 > R_{th}$ | $R_3$ |
| $R'_1 < -R_{th}$ | $R_4$ |
| $R'_5 > R_{th}$ | $R_5$ |
| $R'_3 < -R_{th}$ | $R_6$ |
| $R'_7 > R_{th}$ | $R_7$ |
| $R'_5 < -R_{th}$ | $R_8$ |

The control methodology presented above looks promising and we are anxious to able to test it using the actual robot hardware. It appears that it is a general algorithm and should be easily extensible to arbitrary sets of thrusters including those out of the plane (i.e. the 3-D case).

## 4.6 Future Work

Now that the arms are nearly complete, our emphasis can shift to the robot's electrical systems which make up the "analog layer" of our modular design. This layer which was described in detail in an earlier report will contain replaceable battery packs, DC-to-DC power converters, and an analog card cage containing sensor and driver electronics, power distribution and control circuitry, and safety monitoring and cutout devices.

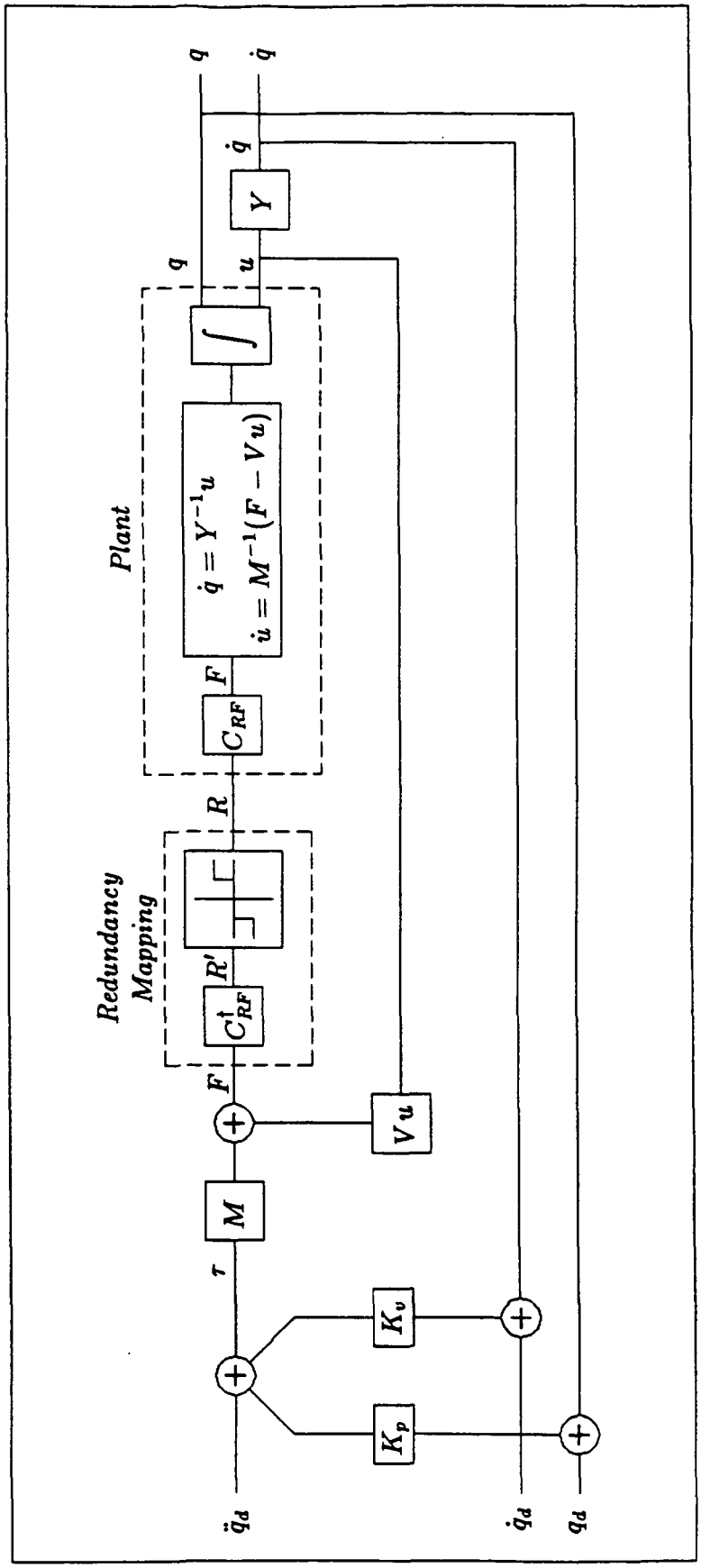



Figure 4.3: Block diagram of computed torque controller with additional thresholding and mapping stages.

~

# Chapter 5

# Locomotion Enhancement via Arm Pushoff (LEAP)

Warren J. Jasper
Roberto Zanutta

## 5.1 Introduction

To perform complex assembly tasks, an autonomous vehicle needs to move from one place to another. The use of propellants may not be ideal because of cost and safety factors. Also, the use of thrusters may disturb the environment by impacting a target which the robot is trying to grasp. Our alternative approach is called LEAP: Locomotion Enhancement via Arm Pushoff. In LEAP, the vehicle pushes itself off from a large space object and "leaps" to the desired resting place or simply "crawls" along an object. This is the common mode of locomotion used by the astronauts while in the Space Shuttle. This new project was added to investigate the problems and issues involved in autonomous space locomotion. The first phase of the project involves: devising the experiment, deriving the equations of motion and candidate control laws, and then simulating the model to size physical parameters for the actual experiment. The second phase encompasses design and fabrication of the vehicle, while the third phase is to experimentally verifiy the theoretical development. The following paragraphs describe the progress onphase one.

## 5.2 The Experiment

A new air-cushion vehicle is being designed to study LEAP. This vehicle should simulate the motions that an autonomous space robot would perform while in the space station or maneuvering out in space. The experiment will consist of the vehicle pushing off a bar located on one side of the granite table, rotating 180 degs, and catching itself by grasping a bar located at the other end of the table. Ideally, one would like to complete this task without the use of thrusters. However, at the point of initial release from the bar, errors

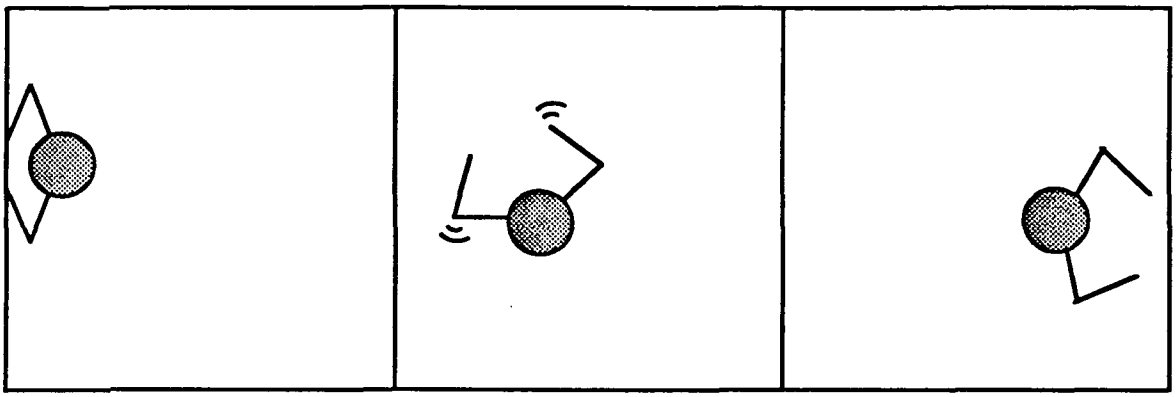


Figure 5.1: The LEAP Demonstration

in the velocity of the center of mass of the vehicle can only be corrected using thrusters. To enhance the robustness of this approach, thrusters will be incorporated into the control laws for midcourse correction. Figure 5.1 shows the robot in three configurations: pushing off the bar, rotating, and catching itself at the other end. By incorporating crawling and leaping, the robot can position itself anywhere on the table with a minimum amount of propellant. This investigation complements current work done at the Stanford Aerospace Robotic Laboratory by incorporating global navigation and object manipulation into a general study of locomotion.

## 5.3 Vehicle Modelling

A great deal of design work has been done under this NASA contract in the field of multiple-arm autonomous vehicles. Because the underlying philosophy of our current robots allows for flexibility in design, the new vehicle will need only a few design modifications. These changes include the addition of a momentum wheel subsystem and grippers. Figure 5.2 shows a schematic view of the LEAP robot used for simulation. This vehicle has eight degrees of freedom, one more than the previous design, due to the addition of a momentum wheel.

The momentum wheel is necessary to perform rotation without the use of thrusters. This reduces the propellant cost and provides linear control authority for rotation. Unfortunately, one can not obviate the need for thrusters completely because they are necessary for midcourse corrections and deceleration in free space. Also, a new gripper design is needed which incorporates force sensing and compliance when grasping an object.

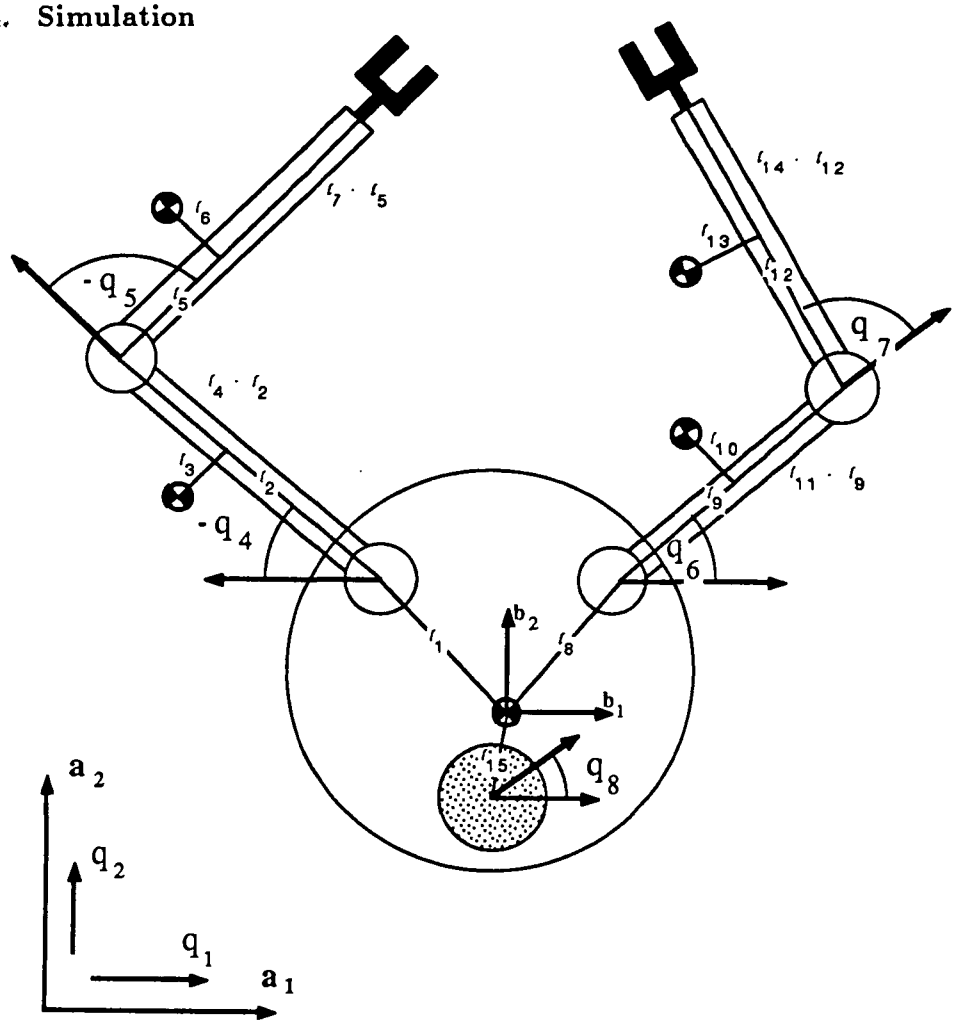



Figure 5.2: Schematic Drawing of the Robot

## 5.4 Simulation

The simulation is used as a design tool to derive specifications for the motors and the sensors as well as evaluating different control schemes. After the initial experiments, a further iteration of the simulator will be done to increase its fidelity and to incorporate second-order effects. Using Kane's method, the equations of motion were derived in a similar manner as those described in Appendix C. These equations were incorporated into the non-linear dynamic simulation. In addition to plotting the various states of the robot, the computer, a Sun 3/160 workstation, can also produce a real-time graphics display. This two dimensional display allows one to understand how the entire system is responding to various control laws.

To simulate the experiment, we implemented the task of leaping from one end of the granite table to the other as a finite state machine by dividing the entire task into ten separate states. In each state, the computer uses the appropriate set of boundary conditions and

control laws to achieve a desired trajectory. The control laws used are proportional derivative control (PD), proportional integral derivative control (PID), and computed torque. These control laws are modified to account for discontinuities in the state due to changing boundary conditions. These conditions occur when the robot lets go of the bar and goes from a nonholonomic closed-loop chain configuration with four degrees of freedom to a holonomic free floating configuration with eight degrees of freedom. A similar condition arises when the robot grasps the bar at the other end of the table, however, there is always the possibility that both arms will not grasp simultaneously, and thus we allow for two additional conditions.

## 5.5 Future Work

With the completion of the first phase, we plan to proceed to the second phase of the project which includes fabrication and integration of the various subsystems into a working vehicle. This phase should last well into the Fall of 1988. Also in the second phase, we will look into new sensor technologyfor force and rate sensing and incorporate these sensors into the vehicle.

# Appendix A

# Accurate Real-time Vision Sensor

Stan Schneider [1]

## A.1 The Vision Sensor Problem

Advanced end-point control of complex systems requires tracking several objects simultaneously. End-point information to be used for control feedback must be highly accurate, and available at a high sampling rate. To provide this information, we are developing a vision system capable of tracking at least four points with an error of less than one part in one thousand. Sample rates will be at least 60 times per second per target. No systems currently commercially available provide sufficient performance. This report describes progress to date in the development of this vision sensor.

### A.1.1 Proposed Solution

Recent advances in CCD camera technology have made pixel-based end point detection feasible. At the current time, 440 by 240 pixel arrays are available with sampling rates greater than 60 Hz. Coupled with local processing power to provide an intelligent interface, this allows sample rates well within the above specifications. Unfortunately, the accuracy, 240:1, is not acceptable for precision control. The problem thus becomes one of gleaning sub-pixel accuracy from the available data.

By utilizing the available gray scale resolution of the camera, this is practicable. Instead of simple binary (white dot or LED) targets, we propose to use targets that vary in reflectivity, from black at the edges to near white in the center. They span an area of approximately 8 by 8 pixels. Thus, each pixel in the 8x8 grid contains information about the location of the center. (In the noise-free case, one could calculate the distance to the center from each pixel exactly.) A simple, fast centroid (center-of-brightness) calculation then yields an estimate of the actual center.

[1] Summary of work begun in 1986/1987 under AFOSR contract. The vision sensor system described herein will be developed under the current NASA contract, and incorporated into the fixed-base cooperative manipulation experiment. This appendix is included for completeness.

### A.1.2 Summary of Progress

A probabilistic analysis has been completed. In addition, a full simulation of the target location procedure was implemented to verify the analysis. Finally, a simple experimental system was constructed, and its performance measured. The results are encouraging. With the new camera that has been ordered, the above performance specification should be achievable.

## A.2 Theoretical Analysis

For the purposes of this discussion, we assume an $N \times N$ pixel sub-grid. This grid is sufficient to encompass the entire target area. The center of the target is calculated as the centroid of illumination. We will calculate the centroid in one dimension only, the other is identical.

Let:

| | | | |
|---|---|---|---|
| $p_{ij}$ | | | represent the $i, j^{th}$ pixel, |
| S | = | $\sum w_i p_{ij}$ | be the weighted pixel illumination, and |
| T | = | $\sum p_{ij}$ | be the total illumination. |

For our purposes, $w_i$ is simply the distance from the center of the coordinate system, $w_i = i - i_0$.

The centroid is found simply by:

$$Z = \frac{S}{T} \tag{A.1}$$

Quantization effects are insignificant in comparison with the camera noise. Thus, the noise can be assumed to be gaussian, and white. Now, if each pixel is corrupted by independent gaussian noise of variance $\sigma^2$, then both S and T are gaussian random variables with:

$$\sigma_t^2 = \sum_{i,j} \sigma^2 = N^2\sigma^2 \tag{A.2}$$

$$\mu_t = \sum_{i,j} E(p_{ij}) \tag{A.3}$$

$$\sigma_s^2 = \sum_{i,j} (i - i_0)^2\sigma^2 \tag{A.4}$$

$$\mu_s = \sum_{i,j} E((i - i_0)p_{ij}) \tag{A.5}$$

We now develop an expression for the probability density function of the centroid. The distribution function is found by integrating over the region of $s/t \leq z$:

$$F_z(z) = \int_{-\infty}^{0}\int_{zt}^{\infty} f_{ts}(t,s)dsdt + \int_{0}^{\infty}\int_{-\infty}^{zt} f_{ts}(t,s)dsdt \tag{A.6}$$

Where

$$f_{ts}(t,s) = \frac{1}{2\pi\sigma_s\sigma_t} e^{-1/2((\frac{s-\mu_s}{\sigma_s})^2+(\frac{t-\mu_t}{\sigma_t})^2)} \tag{A.7}$$

Using the relation $\int_{-\infty}^{0} g(x)dx = \int_{0}^{\infty} g(-x)dx$ and Liebnitz's rule for differentiation, we obtain:

$$f_z(z) = \frac{d}{dz}F_z(z) = \int_0^\infty t(f_{ts}(t,zt) + f_{ts}(-t,-zt))dt \tag{A.8}$$

Substitution of A.7 yields:

$$f_z(z) = K[\int_0^\infty te^{(At^2-Bt+C)dt} + \int_0^\infty te^{(At^2+Bt+C)dt}] \tag{A.9}$$

with:

$$A = \frac{1}{2\sigma_t^2\sigma_s^2}(\sigma_s^2 + z^2\sigma_t^2) \tag{A.10}$$

$$B = \frac{1}{2\sigma_t^2\sigma_s^2}(2\mu_t\sigma_s^2 + 2z\mu_s\sigma_t^2) \tag{A.11}$$

$$C = \frac{1}{2\sigma_t^2\sigma_s^2}(\mu_t^2\sigma_s^2 + \mu_s^2\sigma_t^2) \tag{A.12}$$

$$K = \frac{1}{2\pi\sigma_t\sigma_s} \tag{A.13}$$

After integration and collection of terms, the density function is:

$$f_z(z) = K[\frac{e^{-C}}{A} + \frac{B}{2A}\sqrt{\frac{\pi}{A}}e^{\frac{(B^2-4AC)}{4A}}\Phi(\frac{B}{2\sqrt{A}})] \tag{A.14}$$

Here, $\Phi(x)$ is the error function,

$$\Phi(x) = \frac{2}{\sqrt{\pi}}\int_0^\infty e^{-x^2}dx$$

Note that further simplifications are possible. For example, $\frac{(B^2-4AC)}{4A}$ can be reduced to $\frac{(\mu_s-z\mu_t)^2}{\sigma_s^2\sigma_t^2}$.

In the zero mean case, $C = B = 0$, this reduces to the well known ratio of gaussian densities, the Cauchy density. An interesting result is that the best estimate of the centroid is not simply the above ratio of $S/T$, but the much more complicated expected value of $f_z(z)$. A rough evaluation of the actual best target center estimate can be made by substituting reasonable values for all the parameters (which yields $\mu_t$ is large, $\mu_s$ is nearly zero, and $\sigma_s$ is significantly larger than $\sigma_t$). This rough analysis yields two results: 1) $S/T$ is a very good estimate of the maximum, and 2) the actual best estimate is always slightly smaller magnitude (closer to zero) than $S/T$.

## A.3 Simulation

To verify the above analysis, a full simulation of the target center calculation was implemented. This program allows rapid variation of such variables as: pixel signal-to-noise ratio, quantization effects, target size, target reflectivity profile, and background illumination variation.

### A.3.1 Simulation Description

The target is modeled as a truncated cone of variable intensity. Figure A.1 depicts a typical target reflectivity profile. All labeled parameters are user-selectable.

Target locations are selected randomly near the center of the grid. The camera CCD response is calculated by integrating the target illumination over each pixel. Gaussian noise is added, and the value quantized to the user-specified number of levels. The target location is then determined by the above algorithm. The program outputs raw data, average error, and an error histogram.

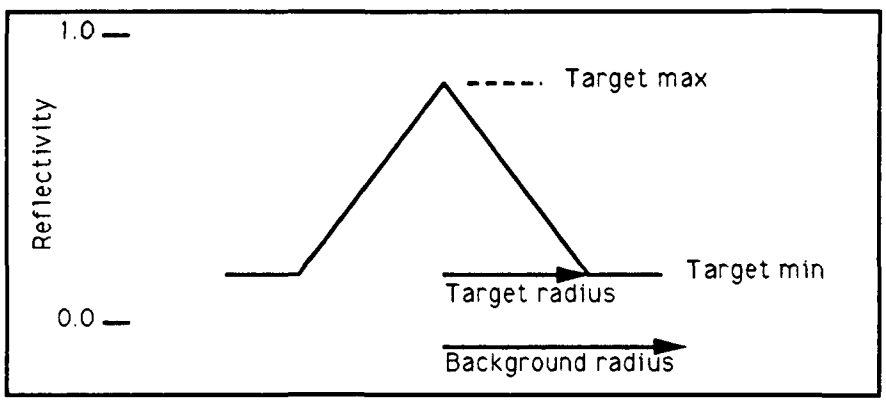



Figure A.1: Simulated Target Reflectivity Profile

### A.3.2 Simulation Results

The graph in figure A.2 shows a plot of the histogram of error values obtained over 1000 trials. The above probability density function is also plotted for comparison. Under realistic conditions (100:1 signal/noise amplitude; 3 pixel radius linear density targets), the method located the center to better than 1/10th of a pixel worst case, 1/20th pixel average case. With a 440x240 pixel camera, this represents very respectable accuracy (typically 1 part in 2000!).

The centroid calculation algorithm could be improved slightly. Due to the division nonlinearity, the best estimate of the true centroid is not simply $S/T$, but a value slightly smaller. Use of a simple scaling factor could reduce the frequency of the larger errors. In the case of this system, it is unlikely this effect will be significant.

## A.4 Preliminary Experimental Results

Before ordering high speed equipment, a preliminary proof-of-concept experiment was conducted. Using an IBM PC with a commercially available low-speed frame digitizer, data from a standard format camera was collected. Figure A.3 shows a representative target.

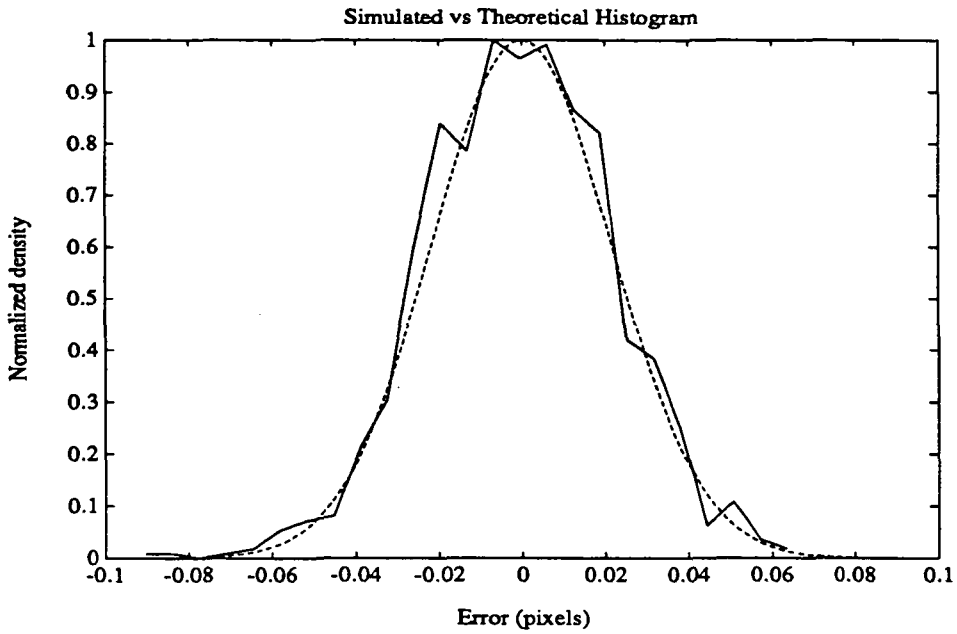



Figure A.2: Simulated Error Histogram

The centroid calculation algorithm was implemented with no attempt at real time performance.

The camera produced 256 lines of data over an 80 inch field of view. Several measured target locations were chosen, and the results analyzed for both stability and accuracy. In all cases, stability was reasonable; jitter was never greater than .05 pixels (.02 inches). Also, the accuracy was acceptable. The center of the target was consistently found with less than a 0.16 pixel (0.05 inches) error. Linearity over larger distances was as good as the lens optics: about 1% over the field of view. Mike Hollars (a student in our lab) has developed a compensation scheme for optical non-linearities that should solve this problem.

## A.5 Summary

In summary, this technology promises significant advantages over the photodiode position sensors currently employed by the lab. This system should provide:

- Multiple target tracking – at least 4 targets.
- High accuracy – at least 1000:1.
- High speed – The system should provide 60+ Hz samples.
- An intelligent interface – freeing the main control processor from low-level sensor data processing.

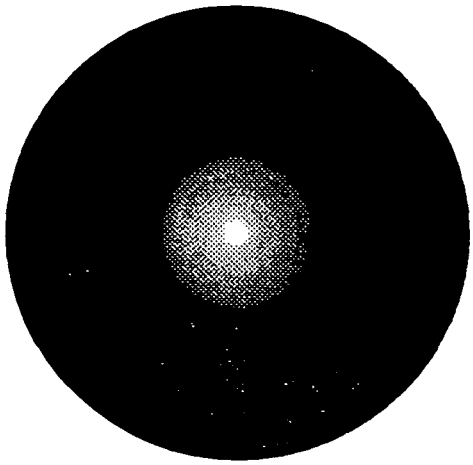


Figure A.3: Example Target

- Future flexibility – Since the entire scene is available to the image processing system, further functionality could be implemented. For instance, objects in the environment could be identified to be used in assemblies, or mapped to be avoided. Use of simple scene analysis could also greatly benefit future cooperative tasks.

# Appendix B

# Serial Chain Manipulator Equations of Motion

Ross Koningstein

This theory for serial chain manipulators is derived using Kane's dynamical analysis techniques. The analysis that follows assumes that the velocities $\mathbf{v}$ of points and angular velocities $\omega$ of bodies in the system under consideration can be expressed in a Newtonian reference frame as follows:

$$\begin{aligned} \mathbf{v}^i &= \sum_{s=1}^{p} \mathbf{v}_s^i u_s \\ \omega^i &= \sum_{s=1}^{p} \omega_s^i u_s \end{aligned}$$

This will be true if no part of the system is undergoing prescribed motions. The partial angular velocities of bodies, and partial velocities of points, as defined by Kane[3], can be shown to be:

$$\begin{aligned} \mathbf{v}_r &= \frac{\partial}{\partial u_r}\mathbf{v} \\ \omega_r &= \frac{\partial}{\partial u_r}\omega \end{aligned}$$

This analysis will show that the elements in the matrix equation

$$\mathbf{M}\dot{u} = -\mathbf{N}u + \mathbf{F}$$

are highly structured and can be computed numerically with a very straightforward algorithm. In order for the algorithm to function, the partial velocities of the first chain element, and the partial angular velocities of all chain elements need to be specified. Manual computation of accelerations, and force components is not required. These derivations assume all velocities, angular velocities, momenta are expressed in a Newtonian reference frame.

## B.1 Structure in Dynamical Equations

In the Kane's Dynamical equations

$$\mathbf{F}_r + \mathbf{F}_r^* = 0$$

The generalized active forces are

$$\mathbf{F}_r = \sum_{\substack{\text{applied} \\ \text{forces i}}} \mathbf{F}^i \cdot \mathbf{v}_r^{i*} + \sum_{\substack{\text{applied} \\ \text{torques j}}} \mathbf{T}^j \cdot \boldsymbol{\omega}_r^j$$

and that the generalized inertia forces are

$$\mathbf{F}_r^* = \sum_{\substack{\text{applied} \\ \text{forces i}}} \mathbf{F}^{i*} \cdot \mathbf{v}_r^{i*} + \sum_{\substack{\text{applied} \\ \text{torques j}}} \mathbf{T}^{j*} \cdot \boldsymbol{\omega}_r^j$$

The generalized inertia force expression, when derived for serial chains, will be examined more closely. Expressions for the inertial forces will be derived using the linear and angular momenta of the bodies in the system. First, the terms due to changes in linear momentum will be examined, then terms due to changes in angular momentum will be examined.

The linear momentum of the $i^{th}$ body is defined as

$$\begin{aligned} \mathbf{L}^i &= m_i \mathbf{v}^{i*} \\ &= m^i \sum_{s=1}^{n} \mathbf{v}_s^{i*} u_s \\ &= \sum_{s=1}^{n} m^i \mathbf{v}_s^{i*} u_s \\ &= \sum_{s=1}^{n} \mathbf{L}_s^i u_s \end{aligned}$$

where the partial linear momentum is defined by

$$\mathbf{L}_s^i \triangleq m_i \mathbf{v}_s^{i*}$$

The inertia force caused by the acceleration of the center of mass of the $i^{th}$ body is:

$$\begin{aligned} \mathbf{F}^{i*} &= -\frac{d}{dt}\mathbf{L}^i \\ &= -\sum_{s=1}^{n} \dot{\mathbf{L}}_s^i u_s - \sum_{s=1}^{n} \mathbf{L}_s^i \dot{u}_s \end{aligned}$$

Its contribution to the generalized inertia forces is

$$\mathbf{F}^{i*} \cdot \mathbf{v}_r^i = -\sum_{s=1}^{n} \dot{\mathbf{L}}_s^i \cdot \mathbf{v}_r^{i*} u_s - \sum_{s=1}^{n} \mathbf{L}_s^i \cdot \mathbf{v}_r^{i*} \dot{u}_s$$

The contribution of the changes in angular momentum will now be examined. The angular momentum of the $i^{th}$ body is defined as

$$\begin{aligned}\mathbf{H}^i &= \mathbf{I}^{i/i*}\boldsymbol{\omega}^i \\ &= \mathbf{I}^{i/i*}\sum_{s=1}^{n}\boldsymbol{\omega}_s^i u_s \\ &= \sum_{s=1}^{n}\mathbf{I}^{i/i*}\mathbf{v}_s^i u_s \\ &= \sum_{s=1}^{n}\mathbf{H}_s^i u_s\end{aligned}$$

where the partial angular momentum is defined by

$$\mathbf{H}_s^i \triangleq \mathbf{I}^{i/i*}\boldsymbol{\omega}_s^i$$

The inertia torque caused by the angular acceleration of the $i^{th}$ body is:

$$\begin{aligned}\mathbf{T}^{i*} &= -\frac{d}{dt}\mathbf{H}^i \\ &= -\sum_{s=1}^{n}\dot{\mathbf{H}}_s^i u_s - \sum_{s=1}^{n}\mathbf{H}_s^i \dot{u}_s\end{aligned}$$

Its contribution to the generalized inertia forces is

$$\mathbf{T}^{i*}\cdot\boldsymbol{\omega}_r^i = \sum_{s=1}^{n}\dot{\mathbf{H}}_s^i\cdot\boldsymbol{\omega}_r^i u_s + \sum_{s=1}^{n}\mathbf{H}_s^i\cdot\boldsymbol{\omega}_r^i\dot{u}_s$$

The generalized inertia forces can be then be expressed as

$$\begin{aligned}\mathbf{F}_r^* &= -\sum_{\substack{\text{All}\\ \text{Bodies}i}}\left(\sum_{s=1}^{n}\dot{\mathbf{L}}_s^i\cdot\mathbf{v}_r^{i*}u_s + \sum_{s=1}^{n}\mathbf{L}_s^i\cdot\mathbf{v}_r^{i*}\dot{u}_s\right) \\ &= -\sum_{\substack{\text{All}\\ \text{Bodies}i}}\left(\sum_{s=1}^{n}\dot{\mathbf{H}}_s^i\cdot\boldsymbol{\omega}_r^i u_s + \sum_{s=1}^{n}\mathbf{H}_s^i\cdot\boldsymbol{\omega}_r^i\dot{u}_s\right)\end{aligned}$$

The generalized inertia force can be separated into two components, one encompassing the derivatives of the generalized speeds, and the other comprising all the rest.

$$\begin{aligned}\mathbf{F}_r^{\mathcal{M}} &\triangleq -\sum_{\substack{\text{All}\\ \text{Bodies}i}}\sum_{s=1}^{n}\mathbf{L}_s^i\cdot\mathbf{v}_r^{i*}\dot{u}_s - \sum_{s=1}^{n}\mathbf{H}_s^i\cdot\boldsymbol{\omega}_r^i\dot{u}_s \\ \mathbf{F}_r^{\mathcal{N}} &\triangleq -\sum_{\substack{\text{All}\\ \text{Bodies}i}}\sum_{s=1}^{n}\dot{\mathbf{L}}_s^i\cdot\mathbf{v}_r^{i*}u_s - \sum_{s=1}^{n}\dot{\mathbf{H}}_s^i\cdot\boldsymbol{\omega}_r^i u_s\end{aligned}$$

The inertia and momentum scalars which make up the mass matrix and non-linear coupling matrix can then be evaluated as follows:

$$m_{rs} \triangleq \sum_{\substack{\text{All} \\ \text{Bodies i}}} m_i \mathbf{v}_s^{i*} \cdot \mathbf{v}_r^{i*} + \left(\mathbf{I}^{i/i*}\omega_s^i\right) \cdot \omega_r^i$$

$$n_{rs} \triangleq \sum_{\substack{\text{All} \\ \text{Bodies i}}} m_i \dot{\mathbf{v}}_s^{i*} \cdot \mathbf{v}_r^{i*} + \left(\omega^i \times \mathbf{I}^{i/i*}\omega_s^i\right) \cdot \omega_r^i$$

The generalized inertia forces can be constructed as follows:

$$\mathbf{F}_r^* = \mathbf{F}_r^{\mathcal{M}} + \mathbf{F}_r^{\mathcal{N}}$$

where

$$\mathbf{F}_r^{\mathcal{M}} = -\sum_{s=1}^{n} m_{rs}\dot{u}_s$$

$$\mathbf{F}_r^{\mathcal{N}} = -\sum_{s=1}^{n} n_{rs}u_s$$

The mass matrix and non-linear coupling matrix can then be used to express the equations of motion of the system as:

$$\mathbf{M}\dot{u} = -\mathbf{N}u + \mathbf{F}$$

where $\mathbf{F}$ is the generalized active force vector which accounts for the effects of external forces and torques applied to points and bodies in the system:

$$\mathbf{F}_r \triangleq \sum_{\substack{\text{All external} \\ \text{forces}}} \mathbf{v}_r^j \cdot \mathbf{F}^j + \sum_{\substack{\text{All external} \\ \text{torques}}} \omega_r^i \cdot \mathbf{T}^i$$

## B.2 Kinetic Energy and Power Input

In a system as discussed in the previous subsections, the kinetic energy can be expressed as:

$$\begin{aligned}\mathbf{K} &= \frac{1}{2}\sum_{\substack{\text{all} \\ \text{bodies}i}} \left(m^i \mathbf{v}^{\text{c.m.}^2} + \omega^i \cdot \mathbf{I}^{i/i*} \cdot \omega^i\right) \\ &= \frac{1}{2}\sum_{\substack{\text{all} \\ \text{bodies}i}} m^i \mathbf{v}^{i*} \cdot \mathbf{v}^{i*} \\ &\quad + \frac{1}{2}\sum_{\substack{\text{all} \\ \text{bodies}i}} \omega^i \cdot \mathbf{I}^{i/i*} \cdot \omega^i\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{2}\sum_{\substack{\text{all}\\ \text{bodies}i}} m^i \left(\sum_{r=1}^{n} \mathbf{v}_r^{i*} u_r\right) \cdot \left(\sum_{s=1}^{n} \mathbf{v}_s^{i*} u_s\right) \\
&\quad + \frac{1}{2}\sum_{\substack{\text{all}\\ \text{bodies}i}} \left(\sum_{r=1}^{n} \omega_r^i u_r\right) \cdot \mathbf{I}^{i/i*} \cdot \left(\sum_{s=1}^{n} \omega_s^i u_s\right) \\
&= \frac{1}{2}\sum_{\substack{\text{all}\\ \text{bodies}i}} \sum_{r=1}^{n}\sum_{s=1}^{n} m^i \mathbf{v}_r^{i*} \cdot \mathbf{v}_s^{i*} u_r u_s \\
&\quad + \frac{1}{2}\sum_{\substack{\text{all}\\ \text{bodies}i}} \sum_{r=1}^{n}\sum_{s=1}^{n} \omega_r^i \cdot \mathbf{I}^{i/i*} \cdot \omega_s^i u_r u_s \\
&= \frac{1}{2}\sum_{\substack{\text{all}\\ \text{bodies}i}} \sum_{r=1}^{n}\sum_{s=1}^{n} m_{rs} u_r u_s \\
&= u^T \mathbf{M} u
\end{aligned}
$$

The power input to the system as discussed is due to work done by the actuators (arm torquers), and forces exerted on the manipulator endpoints during contact with external objects and can be expressed as follows:

$$
\begin{aligned}
\mathbf{P} &= \sum_{\substack{\text{Applied}\\ \text{Forces}}} \mathbf{F} \cdot \mathbf{v} + \sum_{\substack{\text{Applied}\\ \text{Torques}}} \mathbf{T} \cdot \omega \\
&= \sum_{\substack{\text{Applied}\\ \text{Forces}}} \mathbf{F} \cdot \sum_{r=1}^{n} \mathbf{v}_r u_r + \sum_{\substack{\text{Applied}\\ \text{Torques}}} \mathbf{T} \cdot \sum_{r=1}^{n} \omega_r \\
&= \sum_{r=1}^{n} \left( \sum_{\substack{\text{Applied}\\ \text{Forces}}} \mathbf{F} \cdot \mathbf{v}_r u_r + \sum_{\substack{\text{Applied}\\ \text{Torques}}} \mathbf{T} \cdot \sum_{r=1}^{n} \omega_r \right) \\
&= \sum_{r=1}^{n} \mathbf{F}_r u_r \\
&= \mathbf{F} u
\end{aligned}
$$

## B.3 Formulation of Equations of Motion for Planar Serial Link Manipulators

A system $\mathcal{S}$, consisting of a set of serially connected rigid links in planar configuration has a simple relationship relating link endpoint velocity to link basepoint velocity:

$$\mathbf{v}^{\text{endpoint}} = \mathbf{v}^{\text{basepoint}} + \omega^{\text{link}} \times \mathbf{r}^{\text{basepoint to endpoint}}$$

if this is expressed using partial velocities,

$$\mathbf{v}_r^{\text{endpoint}} = \mathbf{v}_r^{\text{basepoint}} + \omega_r^{\text{link}} \times \mathbf{r}^{\text{basepoint to endpoint}}$$

If we define unit vector $\mathbf{x_i}$ to be along the link, from basepoint to endpoint, and unit vector $\mathbf{y_i}$ to be perpendicular to $\mathbf{x_i}$ and in the plane of the manipulator, then we can define $\mathbf{z}$, a unit vector perpendicular to the plane of the manipulator as:

$$\mathbf{z} \triangleq \mathbf{x_i} \times \mathbf{y_i}$$

Vector $\mathbf{z}$ is the same for all links $i$. Now, the generalized speeds $1..n$ and the angular velocities are related as follows:

$$\omega^{\text{link i}} = u_i \mathbf{z}$$

and the endpoint to basepoint velocity relation for link $i$ of length $l_i$ becomes:

$$\mathbf{r}^{\text{start to end i}} = l_i \, \mathbf{x_i}$$

and the partial velocity relations along the chain from start $i = 1$ to end $n$, are as follows:

$$\mathbf{v}_r^{\text{endi}} = \begin{cases} \mathbf{v}_r^{\text{starti}} & \text{for i} = 1 \ldots \text{r} - 1 \\ l_i \mathbf{x_i} & \text{for i} = \text{r} \\ 0 & \text{for i} = \text{r} + 1 \ldots \text{n} \end{cases}$$

$$\omega_r^i = \begin{cases} \mathbf{z} & \text{for i} = 1 \ldots \text{r} - 1 \\ 0 & \text{for i} \neq \text{r} \end{cases}$$

Two additional generalized speeds, $u_{n+1,n+2}$ describe the motion of a designated point one one of the chain links, for instance, the center of mass of the heaviest body, the 'robot body'. In order to calculate the partial velocities of the mass center of each link, we define an extra coordinate set, $\mathbf{x_i^*}$, $\mathbf{y_i^*}$, and $\mathbf{z_i}$, such that:

$$\begin{aligned} \mathbf{r}^{\text{start to endi}} &= l_i^* \, \mathbf{x_i} \\ \mathbf{z} &= \mathbf{x_i^*} \times \mathbf{y_i^*} \\ \mathbf{v}^{\text{endpoint}} &= \mathbf{v}^{\text{basepoint}} + \omega^{\text{link}} \times \mathbf{r}^{\text{basepoint to endpoint}} \end{aligned}$$

then, if expressed using partial velocities,

$$\mathbf{v}_r^{\text{endpoint}} = \mathbf{v}_r^{\text{basepoint}} + \omega_r^{\text{link}} \times \mathbf{r}^{\text{basepointtoendpoint}}$$

$$\mathbf{v}_r^{i*} = \begin{cases} \mathbf{v}_r^{starti} & \text{for i} = 1 \ldots \text{r} - 1 \\ l_i \mathbf{x_i} & \text{for i} = \text{r} \\ 0 & \text{for i} = \text{r} + 1 \ldots \text{n} \end{cases}$$

The contribution of body $i$ to the inertia scalar $m_{rs}$, an element of the mass matrix M, can be determined as follows:

$$(m_{rs})^i = m^i \, \mathbf{v}_r^{i*} \cdot \mathbf{v}_s^{i*} + \boldsymbol{\omega}_r^{i*} \cdot \mathbf{I}^{i/i*} \cdot \boldsymbol{\omega}_s^{i*}$$

for the planar system, this reduces to

$$(m_{rs})^i = \begin{cases} \sum_{\text{bodies i}} m^i \, \mathbf{v}_r^{i*} \cdot \mathbf{v}_s^{i*} & \text{for i} \neq \text{r} \\ \sum_{\text{bodies i}} m^i \, \mathbf{v}_n^{i*} \cdot \mathbf{v}_n^{i*} + \mathbf{I}_z^{i/i*} & \text{for n} = \text{r} = \text{s} \end{cases}$$

where

$$\mathbf{I}_z^{i/i*} = \mathbf{z} \cdot \mathbf{I}^{i/i*} \cdot \mathbf{z}$$

and

$$\boldsymbol{\omega}^{\text{link}i} = u_i \mathbf{z}$$

the sum of the effects of all bodies, which build a complete inertia scalar.

$$\begin{aligned} \sum_{\substack{\text{bodies i} \\ \text{case r} = \text{s} = \text{j}}} &= m^1 \mathbf{v}_r^{1*} \cdot \mathbf{v}_s^{1*} + m^2 \mathbf{v}_r^{2*} \cdot \mathbf{v}_s^{2*} + \ldots \\ &= m^j \, \mathbf{v}_j^{j*} \cdot \mathbf{v}_j^{j*} + m^j \mathbf{v}_j^{(j+1)*} \cdot \mathbf{v}_j^{(j+1)*} + \ldots \\ &= m^j \left(l^{j*}\right)^2 + m^{j+1} \mathbf{v}_j^{\text{end}_\text{j}} \cdot \mathbf{v}_j^{\text{end}_\text{j}} + \ldots \\ &= m^j \left(l^{j*}\right)^2 \\ &\quad + \left(m^{j+1} + m^{j+3} + \cdots\right) \mathbf{v}_j^{\text{end}_\text{j}} \cdot \mathbf{v}_j^{\text{end}_\text{j}} \\ \text{case r} \neq \text{s}, \text{k} = \max(\text{r},\text{s}) & \\ &= m^k \, l^{j*} \mathbf{y}_\mathbf{k}^* \cdot \mathbf{v}_\mathbf{r}^{\text{start}\mathbf{k}} \\ &\quad + \left(m^{k+1} + m^{k+2} + \cdots\right) \mathbf{v}_j^{\text{end}_\text{k}} \cdot \mathbf{v}_j^{\text{end}_\text{k}} \end{aligned}$$

a similar derivation for the 'non-linear' momentum scalar terms:

$$(n_{rs})^i = m^i \, \mathbf{v}_r^{i*} \cdot \dot{\mathbf{v}}_s^{i*} + \boldsymbol{\omega}_r^i \cdot \dot{\mathbf{H}}^i$$

where

$$\boldsymbol{\omega}_r^{i*} \cdot \dot{\mathbf{H}}^i = 0$$

In the planar configuration, no torques due to changes in orientation of the angular momentum vector of any of the bodies occur.

If we define

$$m^*_{rs} \triangleq m^{k+1} + m^{k+2} + \cdots + m^n$$

where

$$k = \max(r, s)$$

then the inertia and momentum scalars for the mass and non-linear coupling terms matrices can be expressed as:

$$\begin{aligned} m_{rs} \quad = \quad & m^*_{rs}\ \mathbf{v}_r^{\text{end n}} \cdot \mathbf{v}_s^{\text{end n}} \\ & + m^k\ (l^* k \mathbf{y}^*_{\mathbf{k}}) \cdot \mathbf{v}_r^{\text{end n}} \\ \text{and if j = r = s} & \\ & + m^j\,(l_{j*})^2 \\ & + \mathbf{I}_z^{j/j*} \end{aligned}$$

similarly, the expression for the non-linear coupling term matrix scalars is:

$$\begin{aligned} n_{rs} \quad = \quad & m^*_{rs}\ \mathbf{v}_r^{\text{end n}} \cdot \dot{\mathbf{v}}_s^{\text{end n}} \\ \text{if s > r} & \\ & + m^k\ (l^*_k \mathbf{y}^*_{\mathbf{k}})\, \dot{\mathbf{v}}_r^{\text{end n}} \\ \text{if r > s} & \\ & - m^k\ (l^*_k \mathbf{x}^*_{\mathbf{k}} \mathbf{u}_{\mathbf{k}})\, \mathbf{v}_r^{\text{end n}} \end{aligned}$$

## B.4 Control Specification Equations

This section deals with the means whereby control specifications can be expressed in terms of quantities expressed or derived in the previous chapters. We wish to be able to determine the Jacobian scalars $j_{rs}$ which form a Jacobian matrix of the form:

$$\mathbf{v}^{\text{endpoint}} = \mathbf{J} u_{1..n+2}$$

The endpoint velocity can be expressed in terms of its partials as:

$$\mathbf{v}^{\text{endpoint}} = \sum_{r=1}^{n+2} \mathbf{v}_r^{\text{endpoint}} u_r$$

and therefore endpoint velocity can be expressed in terms of speeds along the inertial 'x' and 'y' directions, that is, along unit vectors **x** and **y**:

$$\mathbf{v}^{\text{x endpoint}} = \sum_{r=1}^{n+2} \mathbf{v}_r^{\text{endpoint}} \cdot \mathbf{x}\, u_r$$
$$\mathbf{v}^{\text{y endpoint}} = \sum_{r=1}^{n+2} \mathbf{v}_r^{\text{endpoint}} \cdot \mathbf{y}\, u_r$$

the elements of the Jacobian are therefore:

$$j_{1s} = \mathbf{v}_r^{\text{endpoint}} \cdot \mathbf{x}$$
$$j_{2s} = \mathbf{v}_r^{\text{endpoint}} \cdot \mathbf{y}$$

if the endpoint is at the tip of the manipulator, then these partial velocities are exactly the same as the ones used in the dynamical derivations.

The control specification, assumed here to be an acceleration specification for the endpoint, is determined from the following expression:

$$\mathbf{a}^{\text{endpoint}} = \mathbf{J}\dot{u}_{1..n+2} + \dot{\mathbf{J}}u_{1..n+2}$$

the derivative of the Jacobian can be determined from quantities used in the dynamical equation formulation:

$$\dot{j}_{1r} = \dot{\mathbf{v}}_r^{\text{endpoint}} \cdot \mathbf{x}$$
$$\dot{j}_{2r} = \dot{\mathbf{v}}_r^{\text{endpoint}} \cdot \mathbf{y}$$

## B.5 Nonholonomic Constraints: Closed Chain

In a dynamical system with nonholonomic constraints, the generalized speeds $u_{1..n}$ are not independent, rather, one (or more) are dependent on the rest. In the system considered, a serial chain of bodies, this condition can arise when the two ends of the chain touch and are held together, either by a pin joint, or rigidly. The case of a pin constraint on the two dimensional manipulator, a nonholonomic constraint situation, will be analyzed, and the constraint equations will be expressed in terms of quantities used in the derivations for dynamical analysis.

The constraint of endpoint closure is described by:

$$\mathbf{v}^{\text{start}_1} = \mathbf{v}^{\text{end}_n}$$

expanding this into partial velocities,

$$\sum_{r=1}^{n} \mathbf{v}_r^{\text{start}_1} u_r = \sum_{r=1}^{n} \mathbf{v}_r^{\text{end}_n} u_r$$

It is evident that by dot multiplication with two vectors, this vector equation can be reduced to two scalar equations, for the two-dimensional manipulator we are studying. If we choose the vectors such that each is perpendicular to the partial velocity of the end bodies, then the two equations decouple and the last two generalized speeds can be solved for explicitly. Two candidate vectors to accomplish this are the derivatives of the partial velocity vectors of the last two links: $\dot{\mathbf{v}}_n^{\text{end}_n}$ and $\dot{\mathbf{v}}_{n-1}^{\text{end}_{n-1}}$, since these vectors are guaranteed to be perpendicular to their respective partial velocities. This is proven by the fact that

$$\dot{\mathbf{v}}_r^{\text{body i}} = \boldsymbol{\omega}^{\text{body i}} \times \mathbf{v}_r^{\text{body i}}$$

If we use these two vectors to separate the vector constraint equation into two scalar constraint equations, we get:

$$\sum_{r=1}^{n} \mathbf{v}_r^{\text{start}_1} \cdot \dot{\mathbf{v}}_r^{\text{body i}} u_r \;=\; \sum_{r=1}^{n} \mathbf{v}_r^{\text{end}_n} \cdot \dot{\mathbf{v}}_r^{\text{body i}} u_r$$

we note that

$$\begin{aligned} \mathbf{v}_{1..n}^{\text{start}_1} &= 0 \\ \mathbf{v}_{1..n}^{\text{start}_1} - \mathbf{v}_{1..n}^{\text{end}_n} &= 0 \end{aligned}$$

so that the two constraint equations are:

$$\begin{aligned} 0 &= \sum_{r=1}^{n} \mathbf{v}_r^n \cdot \dot{\mathbf{v}}_1^1 u_r \\ 0 &= \sum_{r=1}^{n} \mathbf{v}_r^n \cdot \dot{\mathbf{v}}_n^n u_r \end{aligned}$$

These equations are decoupled for $u_1$ and $u_n$ and can be rearranged to pick out these terms explicitly:

$$\begin{aligned} \mathbf{v}_r^n \cdot \dot{\mathbf{v}}_1^1 &= -\sum_{r=2}^{n-1} \mathbf{v}_r^n \cdot \dot{\mathbf{v}}_1^1 \\ \mathbf{v}_r^n \cdot \dot{\mathbf{v}}_n^n &= -\sum_{r=2}^{n-1} \mathbf{v}_r^n \cdot \dot{\mathbf{v}}_n^n \end{aligned}$$

such that the reduced set of independent speeds, $\tilde{u}_{1..n}$, can be expressed as:

$$\tilde{u}_{1..n} = \mathbf{C} u_{1..n+2}$$

where

$$\left.\begin{array}{rll} c_{1r} & \triangleq & -\frac{\sum_{r=2}^{n-1} \mathbf{v}_r^n \cdot \dot{\mathbf{v}}_1^1}{\mathbf{v}_n^n \cdot \dot{\mathbf{v}}_1^1} \\ c_{nr} & \triangleq & -\frac{\sum_{r=2}^{n-1} \mathbf{v}_r^n \cdot \dot{\mathbf{v}}_n^n}{\mathbf{v}_1^1 \cdot \dot{\mathbf{v}}_n^n} \\ \text{if } j-1=r & c_{jr}=1 & \end{array}\right\} \quad \text{for } r = 1..n$$

$$\left.\begin{array}{rl} \text{if } j-2=r & c_{jr}=1 \\ & c_{jr}=0 \end{array}\right\} \quad \begin{array}{l} \text{for } r=n+1, n+2 \\ \text{for all other } j, r \end{array}$$

The constrained equations of motion can then be formulated as follows:

$$\tilde{\mathbf{M}}\tilde{u}_{1..n} = -\tilde{\mathbf{N}}u_{1..n+2} + \tilde{\mathbf{F}}_{1..n}$$

where

$$\begin{aligned} \tilde{\mathbf{M}} &\triangleq \mathbf{C}\mathbf{M}\mathbf{C}^T \\ \tilde{\mathbf{N}} &\triangleq \mathbf{C}\left(\mathbf{N} - \mathbf{M}\dot{\mathbf{C}}^T\right) \\ \tilde{\mathbf{F}} &\triangleq \mathbf{C}\mathbf{F} \end{aligned}$$

~

# Appendix C

# Derivations of the Equations of Motion for LEAP Vehicle

Warren Jasper. J

## C.1 Definitions of the Generalized Speeds

Using Kane's method for deriving the equations of motion, the following list defines the generalized speeds.

$$
\begin{aligned}
u_1 &\triangleq {}^A\boldsymbol{v}^{B^*} \cdot \boldsymbol{b}_1 = \dot{q}_1 \cos(q_3) + \dot{q}_2 \sin(q_3) \\
u_2 &\triangleq {}^A\boldsymbol{v}^{B^*} \cdot \boldsymbol{b}_2 = \dot{q}_2 \cos(q_3) - \dot{q}_1 \sin(q_3) \\
u_3 &\triangleq {}^A\boldsymbol{\omega}^{B} \cdot \boldsymbol{k} = \dot{q}_3 \\
u_4 &\triangleq {}^A\boldsymbol{\omega}^{C} \cdot \boldsymbol{k} = \dot{q}_3 + \dot{q}_4 \\
u_5 &\triangleq {}^A\boldsymbol{\omega}^{D} \cdot \boldsymbol{k} = \dot{q}_3 + \dot{q}_4 + \dot{q}_5 \\
u_6 &\triangleq {}^A\boldsymbol{\omega}^{E} \cdot \boldsymbol{k} = \dot{q}_3 + \dot{q}_6 \\
u_7 &\triangleq {}^A\boldsymbol{\omega}^{F} \cdot \boldsymbol{k} = \dot{q}_3 + \dot{q}_6 + \dot{q}_7 \\
u_8 &\triangleq {}^A\boldsymbol{\omega}^{G} \cdot \boldsymbol{k} = \dot{q}_3 + \dot{q}_8
\end{aligned}
$$

Using these definitions for the generalized speed, the velocities of the points of interest become:

$$
\begin{aligned}
{}^A\boldsymbol{v}^{B^*} &= u_1\,\boldsymbol{b}_1 + u_2\,\boldsymbol{b}_2 && \text{(C.1)} \\
{}^A\boldsymbol{v}^{P_3} &= u_1\,\boldsymbol{b}_1 + u_2\,\boldsymbol{b}_2 + u_3\,l_1\,\boldsymbol{r}_2 && \text{(C.2)} \\
{}^A\boldsymbol{v}^{C^*} &= u_1\,\boldsymbol{b}_1 + u_2\,\boldsymbol{b}_2 + u_3\,l_1\,\boldsymbol{r}_2 + u_4\,l_2\,\boldsymbol{c}_2 - u_4\,l_3\,\boldsymbol{c}_1 && \text{(C.3)} \\
{}^A\boldsymbol{v}^{P_4} &= u_1\,\boldsymbol{b}_1 + u_2\,\boldsymbol{b}_2 + u_3\,l_1\,\boldsymbol{r}_2 + u_4\,l_4\,\boldsymbol{c}_2 && \text{(C.4)}
\end{aligned}
$$

$$
\begin{aligned}
{}^A\boldsymbol{v}^{D^*} &= u_1\,\boldsymbol{b}_1 + u_2\,\boldsymbol{b}_2 + u_3\,l_1\,\boldsymbol{r}_2 + u_4\,l_4\,\boldsymbol{c}_2 + u_5\,l_5\,\boldsymbol{d}_2 - u_5\,l_6\,\boldsymbol{d}_1 && \text{(C.5)}\\
{}^A\boldsymbol{v}^{P_5} &= u_1\,\boldsymbol{b}_1 + u_2\,\boldsymbol{b}_2 + u_3\,l_1\,\boldsymbol{r}_2 + u_4\,l_4\,\boldsymbol{c}_2 + u_5\,l_7\,\boldsymbol{d}_2 && \text{(C.6)}\\
{}^A\boldsymbol{v}^{P_6} &= u_1\,\boldsymbol{b}_1 + u_2\,\boldsymbol{b}_2 + u_3\,l_8\,\boldsymbol{r}_4 && \text{(C.7)}\\
{}^A\boldsymbol{v}^{E^*} &= u_1\,\boldsymbol{b}_1 + u_2\,\boldsymbol{b}_2 + u_3\,l_8\,\boldsymbol{r}_4 + u_6\,l_9\,\boldsymbol{e}_2 - u_6\,l_{10}\,\boldsymbol{e}_1 && \text{(C.8)}\\
{}^A\boldsymbol{v}^{P_7} &= u_1\,\boldsymbol{b}_1 + u_2\,\boldsymbol{b}_2 + u_3\,l_8\,\boldsymbol{r}_4 + u_6\,l_{11}\,\boldsymbol{e}_2 && \text{(C.9)}\\
{}^A\boldsymbol{v}^{F^*} &= u_1\,\boldsymbol{b}_1 + u_2\,\boldsymbol{b}_2 + u_3\,l_8\,\boldsymbol{r}_4 + u_6\,l_{11}\,\boldsymbol{e}_2 + u_7\,l_{12}\,\boldsymbol{f}_2 - u_7\,l_{13}\,\boldsymbol{f}_1 && \text{(C.10)}\\
{}^A\boldsymbol{v}^{P_8} &= u_1\,\boldsymbol{b}_1 + u_2\,\boldsymbol{b}_2 + u_3\,l_8\,\boldsymbol{r}_4 + u_6\,l_{11}\,\boldsymbol{e}_2 + u_7\,l_{14}\,\boldsymbol{f}_2 && \text{(C.11)}\\
{}^A\boldsymbol{v}^{G^*} &= u_1\,\boldsymbol{b}_1 + u_2\,\boldsymbol{b}_2 + u_3\,l_{15}\,\boldsymbol{r}_6 && \text{(C.12)}
\end{aligned}
$$

### C.1.1 Accelerations

By differentiating the above velocities with respect to time in the reference frame A, the following eight accelerations of interest become:

$$
\begin{aligned}
{}^A\boldsymbol{a}^{B^*} &= \dot{u}_1\,\boldsymbol{b}_1 + \dot{u}_2\,\boldsymbol{b}_2 - u_2u_3\,\boldsymbol{b}_1 + u_1u_3\,\boldsymbol{b}_2\\
{}^A\boldsymbol{a}^{C^*} &= \dot{u}_1\,\boldsymbol{b}_1 + \dot{u}_2\,\boldsymbol{b}_2 - u_2u_3\,\boldsymbol{b}_1 + u_1u_3\,\boldsymbol{b}_2 + \dot{u}_3l_1\,\boldsymbol{r}_2 + \dot{u}_4l_2\,\boldsymbol{c}_2 -\\
&\quad \dot{u}_4l_3\,\boldsymbol{c}_1 - u_3{}^2l_1\,\boldsymbol{r}_1 - u_4{}^2l_2\,\boldsymbol{c}_1 - u_4{}^2l_3\,\boldsymbol{c}_2\\
{}^A\boldsymbol{a}^{D^*} &= \dot{u}_1\,\boldsymbol{b}_1 + \dot{u}_2\,\boldsymbol{b}_2 - u_2u_3\,\boldsymbol{b}_1 + u_1u_3\,\boldsymbol{b}_2 + \dot{u}_3l_1\,\boldsymbol{r}_2 + \dot{u}_4l_4\,\boldsymbol{c}_2 +\\
&\quad \dot{u}_5l_5\,\boldsymbol{d}_2 - \dot{u}_5l_6\,\boldsymbol{d}_1 - u_3{}^2l_1\,\boldsymbol{r}_1 - u_4{}^2l_4\,\boldsymbol{c}_1 - u_5{}^2l_5\,\boldsymbol{d}_1 - u_5{}^2l_6\,\boldsymbol{d}_2\\
{}^A\boldsymbol{a}^{E^*} &= \dot{u}_1\,\boldsymbol{b}_1 + \dot{u}_2\,\boldsymbol{b}_2 - u_2u_3\,\boldsymbol{b}_1 + u_1u_3\,\boldsymbol{b}_2 + \dot{u}_3l_8\,\boldsymbol{r}_4 + \dot{u}_6l_9\,\boldsymbol{e}_2 -\\
&\quad \dot{u}_6l_{10}\,\boldsymbol{e}_1 - u_3{}^2l_8\,\boldsymbol{r}_3 - u_6{}^2l_9\,\boldsymbol{e}_1 - u_6{}^2l_{10}\,\boldsymbol{e}_2\\
{}^A\boldsymbol{a}^{F^*} &= \dot{u}_1\,\boldsymbol{b}_1 + \dot{u}_2\,\boldsymbol{b}_2 - u_2u_3\,\boldsymbol{b}_1 + u_1u_3\,\boldsymbol{b}_2 + \dot{u}_3l_8\,\boldsymbol{r}_4 + \dot{u}_6l_{11}\,\boldsymbol{e}_2 +\\
&\quad \dot{u}_7l_{12}\,\boldsymbol{f}_2 - \dot{u}_7l_{13}\,\boldsymbol{f}_1 - u_3{}^2l_8\,\boldsymbol{r}_3 - u_6{}^2l_{11}\,\boldsymbol{e}_1 - u_7{}^2l_{12}\,\boldsymbol{f}_1 - u_7{}^2l_{13}\,\boldsymbol{f}_2\\
{}^A\boldsymbol{a}^{G^*} &= \dot{u}_1\,\boldsymbol{b}_1 + \dot{u}_2\,\boldsymbol{b}_2 - u_2u_3\,\boldsymbol{b}_1 + u_1u_3\,\boldsymbol{b}_2 + \dot{u}_3l_{15}\,\boldsymbol{r}_6 - u_3{}^2l_{15}\,\boldsymbol{r}_5\\
{}^A\boldsymbol{a}^{P_5} &= \dot{u}_1\,\boldsymbol{b}_1 + \dot{u}_2\,\boldsymbol{b}_2 - u_2u_3\,\boldsymbol{b}_1 + u_1u_3\,\boldsymbol{b}_2 + \dot{u}_3l_1\,\boldsymbol{r}_2 + \dot{u}_4l_4\,\boldsymbol{c}_2 +\\
&\quad \dot{u}_5l_7\,\boldsymbol{d}_2 - u_3{}^2l_1\,\boldsymbol{r}_1 - u_4{}^2l_4\,\boldsymbol{c}_1 - u_5{}^2l_7\,\boldsymbol{d}_1\\
{}^A\boldsymbol{a}^{P_8} &= \dot{u}_1\,\boldsymbol{b}_1 + \dot{u}_2\,\boldsymbol{b}_2 - u_2u_3\,\boldsymbol{b}_1 + u_1u_3\,\boldsymbol{b}_2 + \dot{u}_3l_8\,\boldsymbol{r}_4 + \dot{u}_6l_{11}\,\boldsymbol{e}_2 +\\
&\quad \dot{u}_7l_{14}\,\boldsymbol{f}_2 - u_3{}^2l_8\,\boldsymbol{r}_3 - u_6{}^2l_{11}\,\boldsymbol{e}_1 - u_7{}^2l_{14}\,\boldsymbol{f}_1
\end{aligned}
$$

### C.1.2 Partial Velocities and Partial Angular Velocities

The next step in the derivation of the equations of motion is to calculate the partial velocities ${}^A\boldsymbol{v}_r^{P_i}$ and partial angular velocities ${}^A\boldsymbol{\omega}_r^B$ where

$$
{}^A\boldsymbol{v}_r^{P_i} \triangleq \frac{\partial\,{}^A\boldsymbol{v}^{P_i}}{\partial u_r} \qquad \text{(C.13)}
$$

$$
{}^A\boldsymbol{\omega}_r^B \triangleq \frac{\partial\,{}^A\boldsymbol{\omega}^B}{\partial u_r} \qquad \text{(C.14)}
$$

The following is a table of the partial velocities and partial angular velocities for the robot.

| r | ${}^A\boldsymbol{v}_r^{B^*}$ | ${}^A\boldsymbol{v}_r^{P_3}$ | ${}^A\boldsymbol{v}_r^{C^*}$ | ${}^A\boldsymbol{v}_r^{P_4}$ | ${}^A\boldsymbol{v}_r^{D^*}$ | ${}^A\boldsymbol{v}_r^{P_5}$ |
|---|---|---|---|---|---|---|
| 1 | $\boldsymbol{b}_1$ | $\boldsymbol{b}_1$ | $\boldsymbol{b}_1$ | $\boldsymbol{b}_1$ | $\boldsymbol{b}_1$ | $\boldsymbol{b}_1$ |
| 2 | $\boldsymbol{b}_2$ | $\boldsymbol{b}_2$ | $\boldsymbol{b}_2$ | $\boldsymbol{b}_2$ | $\boldsymbol{b}_2$ | $\boldsymbol{b}_2$ |
| 3 | 0 | $l_1\,\boldsymbol{r}_2$ | $l_1\,\boldsymbol{r}_2$ | $l_1\,\boldsymbol{r}_2$ | $l_1\,\boldsymbol{r}_2$ | $l_1\,\boldsymbol{r}_2$ |
| 4 | 0 | 0 | $l_2\,\boldsymbol{c}_2 - l_3\,\boldsymbol{c}_1$ | $l_4\,\boldsymbol{c}_2$ | $l_4\,\boldsymbol{c}_2$ | $l_4\,\boldsymbol{c}_2$ |
| 5 | 0 | 0 | 0 | 0 | $l_5\,\boldsymbol{d}_2 - l_6\,\boldsymbol{d}_1$ | $l_7\,\boldsymbol{d}_2$ |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 0 | 0 | 0 | 0 | 0 | 0 |

| r | ${}^A\boldsymbol{v}_r^{P_6}$ | ${}^A\boldsymbol{v}_r^{E^*}$ | ${}^A\boldsymbol{v}_r^{P_7}$ | ${}^A\boldsymbol{v}_r^{F^*}$ | ${}^A\boldsymbol{v}_r^{P_8}$ | ${}^A\boldsymbol{v}_r^{G^*}$ |
|---|---|---|---|---|---|---|
| 1 | $\boldsymbol{b}_1$ | $\boldsymbol{b}_1$ | $\boldsymbol{b}_1$ | $\boldsymbol{b}_1$ | $\boldsymbol{b}_1$ | $\boldsymbol{b}_1$ |
| 2 | $\boldsymbol{b}_2$ | $\boldsymbol{b}_2$ | $\boldsymbol{b}_2$ | $\boldsymbol{b}_2$ | $\boldsymbol{b}_2$ | $\boldsymbol{b}_2$ |
| 3 | $l_8\,\boldsymbol{r}_4$ | $l_8\,\boldsymbol{r}_4$ | $l_8\,\boldsymbol{r}_4$ | $l_8\,\boldsymbol{r}_4$ | $l_8\,\boldsymbol{r}_4$ | $l_{15}\,\boldsymbol{r}_6$ |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | $l_9\,\boldsymbol{e}_2 - l_{10}\,\boldsymbol{e}_1$ | $l_{11}\,\boldsymbol{e}_2$ | $l_{11}\,\boldsymbol{e}_2$ | $l_{11}\,\boldsymbol{e}_2$ | 0 |
| 7 | 0 | 0 | 0 | $l_{12}\,\boldsymbol{f}_2 - l_{13}\,\boldsymbol{f}_1$ | $l_{14}\,\boldsymbol{f}_2$ | 0 |
| 8 | 0 | 0 | 0 | 0 | 0 | 0 |

Table C.1: The partial velocities

| r | ${}^A\boldsymbol{\omega}_r^B$ | ${}^A\boldsymbol{\omega}_r^C$ | ${}^A\boldsymbol{\omega}_r^D$ | ${}^A\boldsymbol{\omega}_r^E$ | ${}^A\boldsymbol{\omega}_r^F$ | ${}^A\boldsymbol{\omega}_r^G$ |
|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | $\boldsymbol{k}$ | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | $\boldsymbol{k}$ | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | $\boldsymbol{k}$ | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | $\boldsymbol{k}$ | 0 | 0 |
| 7 | 0 | 0 | 0 | 0 | $\boldsymbol{k}$ | 0 |
| 8 | 0 | 0 | 0 | 0 | 0 | $\boldsymbol{k}$ |

Table C.2: The partial angular velocities

### C.1.3 The Mass Matrix

By combining terms, one can put the equations of motion into the form

$$M\,\dot{\mathbf{u}} = \boldsymbol{f}_I + \boldsymbol{K}_1\boldsymbol{\tau} \tag{C.15}$$

$$\dot{\mathbf{q}} = \boldsymbol{T}\boldsymbol{u} \tag{C.16}$$

where the mass matrix $M$ is a symmetric positive definite matrix.

$$M = \begin{bmatrix} z_1 & 0 & z_2 & z_3 s_4 + z_4 c_4 & z_5 s_{45} + z_6 c_{45} & z_7 s_6 + z_8 c_6 & z_9 s_{67} + z_{10} c_{67} & 0 \\ 0 & z_1 & z_{11} & z_4 s_4 - z_3 c_4 & z_6 s_{45} - z_5 c_{45} & z_8 s_6 - z_7 c_6 & z_{10} s_{67} - z_9 c_{67} & 0 \\ z_2 & z_{11} & z_{12} & z_{13} s_4 + z_{14} c_4 & z_{15} s_{45} + z_{16} c_{45} & z_{17} s_6 + z_{18} c_6 & z_{19} s_{67} + z_{20} c_{67} & 0 \\ z_3 s_4 + z_4 c_4 & z_4 s_4 - z_3 c_4 & z_{13} s_4 + z_{14} c_4 & z_{21} & z_{22} s_5 + z_{23} c_5 & 0 & 0 & 0 \\ z_5 s_{45} + z_6 c_{45} & z_6 s_{45} - z_5 c_{45} & z_{15} s_{45} + z_{16} c_{45} & z_{22} s_5 + z_{23} c_5 & z_{24} & 0 & 0 & 0 \\ z_7 s_6 + z_8 c_6 & z_8 s_6 - z_7 c_6 & z_{17} s_6 + z_{18} c_6 & 0 & 0 & z_{25} & z_{26} s_7 + z_{27} c_7 & 0 \\ z_9 s_{67} + z_{10} c_{67} & z_{10} s_{67} - z_9 c_{67} & z_{19} s_{67} + z_{20} c_{67} & 0 & 0 & z_{26} s_7 + z_{27} c_7 & z_{28} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & z_{29} \end{bmatrix} \tag{C.17}$$

Where the $z_i$ are constants and have the following definitions:

$$\begin{aligned}
z_1 &\triangleq m_1 + m_2 + m_3 + m_4 + m_5 + m_6 \\
z_2 &\triangleq -(m_2 + m_3) l_1 \sin(\theta_1) - (m_4 + m_5) l_8 \sin(\theta_2) - m_6 l_{15} \sin(\theta_3) \\
z_3 &\triangleq m_2 l_2 + m_3 l_4 \\
z_4 &\triangleq m_2 l_3 \\
z_5 &\triangleq m_3 l_5 \\
z_6 &\triangleq m_3 l_6 \\
z_7 &\triangleq -(m_4 l_9 + m_5 l_{11}) \\
z_8 &\triangleq -m_4 l_{10} \\
z_9 &\triangleq -m_5 l_{12} \\
z_{10} &\triangleq -m_5 l_{13} \\
z_{11} &\triangleq (m_2 + m_3) l_1 \cos(\theta_1) + (m_4 + m_5) l_8 \cos(\theta_2) + m_6 l_{15} \cos(\theta_3) \\
z_{12} &\triangleq I_1 + (m_2 + m_3) {l_1}^2 + (m_4 + m_5) {l_8}^2 + m_6 {l_{15}}^2 \\
z_{13} &\triangleq m_2 l_1 l_3 \cos(\theta_1) - m_3 l_1 l_4 \sin(\theta_1) - m_2 l_1 l_2 \sin(\theta_1)
\end{aligned}$$

$$
\begin{aligned}
z_{14} &\triangleq -m_2 l_1 l_3 \sin(\theta_1) - m_3 l_1 l_4 \cos(\theta_1) - m_2 l_1 l_2 \cos(\theta_1) \\
z_{15} &\triangleq m_3 l_1 l_6 \cos(\theta_1) - m_3 l_1 l_5 \sin(\theta_1) \\
z_{16} &\triangleq -m_3 l_1 l_6 \sin(\theta_1) - m_3 l_1 l_5 \cos(\theta_1) \\
z_{17} &\triangleq -m_4 l_8 l_{10} \cos(\theta_2) + m_5 l_8 l_{11} \sin(\theta_2) + m_4 l_8 l_9 \sin(\theta_2) \\
z_{18} &\triangleq m_4 l_8 l_{10} \sin(\theta_2) + m_5 l_8 l_{11} \cos(\theta_2) + m_4 l_8 l_9 \cos(\theta_2) \\
z_{19} &\triangleq -m_5 l_8 l_{13} \cos(\theta_2) + m_5 l_8 l_{12} \sin(\theta_2) \\
z_{20} &\triangleq m_5 l_8 l_{13} \sin(\theta_2) + m_5 l_8 l_{12} \cos(\theta_2) \\
z_{21} &\triangleq I_2 + m_2(l_2{}^2 + l_3{}^2) + m_3 l_4{}^2 \\
z_{22} &\triangleq -m_3 l_4 l_6 \\
z_{23} &\triangleq m_3 l_4 l_5 \\
z_{24} &\triangleq I_3 + m_3(l_5{}^2 + l_6{}^2) \\
z_{25} &\triangleq I_4 + m_4(l_9{}^2 + l_{10}{}^2) + m_5 l_{11}{}^2 \\
z_{26} &\triangleq -m_5 l_{11} l_{13} \\
z_{27} &\triangleq m_5 l_{11} l_{12} \\
z_{28} &\triangleq I_5 + m_5(l_{12}{}^2 + l_{13}{}^2) \\
z_{29} &\triangleq I_6
\end{aligned}
$$

The non-linear terms in $u_i$ are found in the vector $\boldsymbol{f}_I$, and with the the definitions of the $z_i$ from above, they become:

$$
\begin{aligned}
\boldsymbol{f}_I(1) &= u_2 u_3 z_1 + u_3{}^2 z_{11} + u_4{}^2(z_4 s_4 - z_3 c_4) + u_5{}^2(z_6 s_{45} - z_5 c_{45}) + \\
&\quad u_6{}^2(z_8 s_6 - z_7 c_6) + u_7{}^2(z_{10} s_{67} - z_9 c_{67}) \\
\boldsymbol{f}_I(2) &= -u_1 u_3 z_1 - u_3{}^2 z_2 - u_4{}^2(z_3 s_4 + z_4 c_4) - u_5{}^2(z_5 s_{45} + z_6 c_{45}) - \\
&\quad u_6{}^2(z_7 s_6 + z_8 c_6) - u_7{}^2(z_9 s_{67} + z_{10} c_{67}) \\
\boldsymbol{f}_I(3) &= -u_1 u_3 z_{11} + u_2 u_3 z_2 + u_4{}^2(z_{14} s_4 - z_{13} c_4) + u_5{}^2(z_{16} s_{45} - z_{15} c_{45}) + \\
&\quad u_6{}^2(z_{18} s_6 - z_{17} c_6) + u_7{}^2(z_{20} s_{67} - z_{19} c_{67}) \\
\boldsymbol{f}_I(4) &= -u_1 u_3(z_4 s_4 - z_3 c_4) + u_2 u_3(z_3 s_4 + z_4 c_4) - u_3{}^2(z_{14} s_4 - z_{13} c_4) + \\
&\quad u_5{}^2(z_{23} s_5 - z_{22} c_5) \\
\boldsymbol{f}_I(5) &= -u_1 u_3(z_6 s_{45} - z_5 c_{45}) + u_2 u_3(z_5 s_{45} + z_6 c_{45}) - u_3{}^2(z_{16} s_{45} - z_{15} c_{45}) - \\
&\quad u_4{}^2(z_{23} s_5 - z_{22} c_5) \\
\boldsymbol{f}_I(6) &= -u_1 u_3(z_8 s_6 - z_7 c_6) + u_2 u_3(z_7 s_6 + z_8 c_6) - u_3{}^2(z_{18} s_6 - z_{17} c_6) + \\
&\quad u_7{}^2(z_{27} s_7 - z_{26} c_7) \\
\boldsymbol{f}_I(7) &= -u_1 u_3(z_{10} s_{67} - z_9 c_{67}) + u_2 u_3(z_9 s_{67} + z_{10} c_{67}) - u_3{}^2(z_{20} s_{67} - z_{19} c_{67}) - \\
&\quad u_6{}^2(z_{27} s_7 - z_{26} c_7) \\
\boldsymbol{f}_I(8) &= 0
\end{aligned}
$$

The vector $K_1\tau$ is given by

$$K_1\tau = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & -1 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \\ \tau_5 \end{bmatrix} \tag{C.18}$$

Where the $\tau_i$ are input torques from the five motors.

The $T$ matrix is

$$T = \begin{bmatrix} \cos(q_3) & -\sin(q_3) & 0 & 0 & 0 & 0 & 0 & 0 \\ \sin(q_3) & \cos(q_3) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

### C.1.4 Non-Holonomic Constraint Equations

When both arms of the robot are grasping the bar, the number of degrees of freedom of the system drop from eight to four. At the tip of each arm, there two non-holonomic or velocity constraints which much be satisfied to insure the closed kinematic-chain configuration. The four constraint equations are:

$$^A\boldsymbol{v}^{P_5} \cdot \boldsymbol{b}_1 = 0 \tag{C.19}$$
$$^A\boldsymbol{v}^{P_5} \cdot \boldsymbol{b}_2 = 0 \tag{C.20}$$
$$^A\boldsymbol{v}^{P_8} \cdot \boldsymbol{b}_1 = 0 \tag{C.21}$$
$$^A\boldsymbol{v}^{P_8} \cdot \boldsymbol{b}_2 = 0 \tag{C.22}$$

Substituting Eqn. (C.6) and Eqn. (C.11) yields the following set of non-holonomic constraint equations

$$\boldsymbol{u}_r = \boldsymbol{A}_{rs}\boldsymbol{u}_s \tag{C.23}$$

$$\begin{bmatrix} u_4 \\ u_5 \\ u_6 \\ u_7 \end{bmatrix} = \begin{bmatrix} \frac{\cos(q_4+q_5)}{l_4 \sin(q_5)} & \frac{\sin(q_4+q_5)}{l_4 \sin(q_5)} & \frac{l_1 \sin(q_4+q_5-\theta_1)}{l_4 \sin(q_5)} \\ -\frac{\cos(q_4)}{l_7 \sin(q_5)} & -\frac{\sin(q_4)}{l_7 \sin(q_5)} & -\frac{l_1 \sin(q_4-\theta_1)}{l_7 \sin(q_5)} \\ -\frac{\cos(q_6+q_7)}{l_{11} \sin(q_7)} & -\frac{\sin(q_6+q_7)}{l_{11} \sin(q_7)} & -\frac{l_8 \sin(q_6+q_7-\theta_2)}{l_{11} \sin(q_7)} \\ \frac{\cos(q_6)}{l_{14} \sin(q_7)} & \frac{\sin(q_6)}{l_{14} \sin(q_7)} & \frac{l_8 \sin(q_6-\theta_2)}{l_{14} \sin(q_7)} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

Differentiating the velocities in Eqn. (C.6) and Eqn. (C.11) with respect to time in reference frame A and dotting the acceleration vectors with orthogonal unit vectors $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ gives the following set of equations which expresses $\dot{\mathbf{u}}_r$ in terms of $\dot{\mathbf{u}}_s, \boldsymbol{u}$ and $\boldsymbol{q}$.

$$\dot{\mathbf{u}}_r = \boldsymbol{A}_{rs} \dot{\mathbf{u}}_s + \boldsymbol{b}_{rs} \tag{C.24}$$

$$\begin{bmatrix} \dot{u}_4 \\ \dot{u}_5 \\ \dot{u}_6 \\ \dot{u}_7 \end{bmatrix} = \begin{bmatrix} \frac{\cos(q_4+q_5)}{l_4 \sin(q_5)} & \frac{\sin(q_4+q_5)}{l_4 \sin(q_5)} & \frac{l_1 \sin(q_4+q_5-\theta_1)}{l_4 \sin(q_5)} \\ -\frac{\cos(q_4)}{l_7 \sin(q_5)} & -\frac{\sin(q_4)}{l_7 \sin(q_5)} & -\frac{l_1 \sin(q_4-\theta_1)}{l_7 \sin(q_5)} \\ -\frac{\cos(q_6+q_7)}{l_{11} \sin(q_7)} & -\frac{\sin(q_6+q_7)}{l_{11} \sin(q_7)} & -\frac{l_8 \sin(q_6+q_7-\theta_2)}{l_{11} \sin(q_7)} \\ \frac{\cos(q_6)}{l_{14} \sin(q_7)} & \frac{\sin(q_6)}{l_{14} \sin(q_7)} & \frac{l_8 \sin(q_6-\theta_2)}{l_{14} \sin(q_7)} \end{bmatrix} \begin{bmatrix} \dot{u}_1 \\ \dot{u}_2 \\ \dot{u}_3 \end{bmatrix}$$

$$+ \begin{bmatrix} \frac{u_1 u_3 \sin(q_4+q_5) - u_2 u_3 \cos(q_4+q_5) - u_3^2 l_1 \cos(q_4+q_5-\theta_1) + u_4^2 l_4 \cos(q_5) + u_5^2 l_7}{l_4 \sin(q_5)} \\ \frac{-u_1 u_3 \sin(q_4) + u_2 u_3 \cos(q_4) + u_3^2 l_1 \cos(q_4-\theta_1) - u_4^2 l_4 - u_5^2 l_7 \cos(q_5)}{l_7 \sin(q_5)} \\ \frac{-u_1 u_3 \sin(q_6+q_7) + u_2 u_3 \cos(q_6+q_7) + u_3^2 l_8 \cos(q_6+q_7-\theta_2) + u_6^2 l_{11} \cos(q_7) + u_7^2 l_{14}}{l_{11} \sin(q_7)} \\ \frac{u_1 u_3 \sin(q_6) - u_2 u_3 \cos(q_6) - u_3^2 l_8 \cos(q_6-\theta_2) - u_6^2 l_{11} - u_7^2 l_{14} \cos(q_7)}{l_{14} \sin(q_7)} \end{bmatrix}$$

With this constraint equation, one can derive the reduced set of the equations of motion by partitioning Eqn. (C.15) and adjoining the constraint equation as follows

$$\begin{bmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_{12} \\ \boldsymbol{M}_{12}^T & \boldsymbol{M}_{22} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{u}}_s \\ \dot{\mathbf{u}}_r \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_{I_s} \\ \boldsymbol{f}_{I_r} \end{bmatrix} + \begin{bmatrix} \boldsymbol{K}_{1_s} \\ \boldsymbol{K}_{1_r} \end{bmatrix} \boldsymbol{\tau} \tag{C.25}$$

$$\begin{aligned} &\left[\boldsymbol{M}_{11} + \boldsymbol{M}_{12}\boldsymbol{A}_{rs} + (\boldsymbol{M}_{12}\boldsymbol{A}_{rs})^T + \boldsymbol{A}_{rs}^T \boldsymbol{M}_{22} \boldsymbol{A}_{rs}\right] \dot{\mathbf{u}}_s = \\ &\left[\boldsymbol{f}_{I_s} + \boldsymbol{A}_{rs}^T \boldsymbol{f}_{I_r} - \boldsymbol{M}_{12}\boldsymbol{b}_{rs} - \boldsymbol{A}_{rs}^T \boldsymbol{M}_{22} \boldsymbol{b}_{rs}\right] + \left[\boldsymbol{K}_{1_s} + \boldsymbol{A}_{rs}^T \boldsymbol{K}_{1_r}\right] \boldsymbol{\tau} \end{aligned} \tag{C.26}$$

### C.1.5 Force Constraint

The is an alternate but equivalent formulation of the equations of motions which do not involve solving for the non-holonomic constraints, but rather impose a force boundary condition at the tips to insure that the velocity at the tips of the arms is zero when the arms are grasping the bar. If Eqn. (C.15) is modified to include forces at the tip we get

$$\boldsymbol{M}\,\dot{\boldsymbol{u}} = \boldsymbol{f}_I + \boldsymbol{K}_1\boldsymbol{\tau} + \boldsymbol{K}_2\boldsymbol{f}_T \tag{C.27}$$

Where $\boldsymbol{f}_T$ is a four vector representing the normal and tangential forces exerted by the bar on the arms at points $\boldsymbol{P}_5$ and $\boldsymbol{P}_8$.

$$\boldsymbol{K}_2 = \begin{bmatrix} \cos(q_4+q_5) & -\sin(q_4+q_5) & -\cos(q_6+q_7) & \sin(q_6+q_7) \\ \sin(q_4+q_5) & \cos(q_4+q_5) & -\sin(q_6+q_7) & -\cos(q_6+q_7) \\ l_1\sin(q_4+q_5-\theta_1) & l_1\cos(q_4+q_5-\theta_1) & -l_8\sin(q_6+q_7-\theta_2) & -l_8\cos(q_6+q_7-\theta_2) \\ -l_4\sin(q_5) & -l_4\cos(q_5) & 0 & 0 \\ 0 & -l_7 & 0 & 0 \\ 0 & 0 & -l_{11}\sin(q_7) & -l_{11}\cos(q_7) \\ 0 & 0 & 0 & -l_{14} \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Rewriting Eqn. (C.23), yields

$$\boldsymbol{A}\boldsymbol{u} = 0 \tag{C.28}$$

$$\begin{bmatrix} 1 & 0 & -l_1\sin(\theta_1) & l_4\sin(q_4) & l_7\sin(q_4+q_5) & 0 & 0 \\ 0 & 1 & l_1\cos(\theta_1) & -l_4\cos(q_4) & -l_7\cos(q_4+q_5) & 0 & 0 \\ 1 & 0 & -l_8\sin(\theta_2) & 0 & 0 & -l_{11}\sin(q_6) & -l_{14}\sin(q_6+q_7) \\ 0 & 1 & l_8\cos(\theta_2) & 0 & 0 & l_{11}\cos(q_6) & l_{14}\cos(q_6+q_7) \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \end{bmatrix} = 0$$

Differentiating Eqn. (C.28) with respect to time in reference frame A yields

$$\boldsymbol{A}\dot{\boldsymbol{u}} = \boldsymbol{b} \tag{C.29}$$

$$\boldsymbol{b} = \begin{bmatrix} u_2u_3 + u_3{}^2l_1\cos(\theta_1) - u_4{}^2l_4\cos(q_4) - u_5{}^2l_7\cos(q_4+q_5) \\ -u_1u_3 + u_3{}^2l_1\sin(\theta_1) - u_4{}^2l_4\sin(q_4) - u_5{}^2l_7\sin(q_4+q_5) \\ u_2u_3 + u_3{}^2l_8\cos(\theta_2) + u_6{}^2l_{11}\cos(q_6) + u_7{}^2l_{14}\cos(q_6+q_7) \\ -u_1u_3 + u_3{}^2l_8\sin(\theta_2) + u_6{}^2l_{11}\sin(q_6) + u_7{}^2l_{14}\sin(q_6+q_7) \end{bmatrix}$$

Using Eqn. (C.27) and Eqn. (C.29), the resultant forces exerted by the bar on the arms at points $\boldsymbol{P}_5$ and $\boldsymbol{P}_8$ are

$$\boldsymbol{f}_T = [\boldsymbol{A}\boldsymbol{M}^{-1}\boldsymbol{K}_2]^{-1}[\boldsymbol{b} - \boldsymbol{A}\boldsymbol{M}^{-1}\boldsymbol{f}_I - \boldsymbol{A}\boldsymbol{M}^{-1}\boldsymbol{K}_1\tau] \tag{C.30}$$

### C.1.6 Some Properties of the Center of Mass

In controlling the robot, there are some properties of the center of mass that are of interest. They are the position, velocity and acceleration of the center of mass.

$$\begin{aligned} \boldsymbol{x}^{\mathrm{CM}} &= q_1\boldsymbol{a}_1 + q_2\boldsymbol{a}_2 + \\ &(z_{11} - z_3c_4 + z_4s_4 - z_5c_{45} + z_6s_{45} - z_7c_6 + z_8s_6 - z_9c_{67} + z_{10}s_{67})/z_1\ \boldsymbol{b}_1 \\ &-(z_2 + z_3s_4 + z_4c_4 + z_5s_{45} + z_6c_{45} + z_7s_6 + z_8c_6 + z_9s_{67} + z_{10}c_{67})/z_1\ \boldsymbol{b}_2 \end{aligned}$$

The velocity of the center of mass is given by:

$$\begin{aligned} {}^A\boldsymbol{v}^{\mathrm{CM}} = {} & (z_1u_1 + z_2u_3 + (z_3s_4 + z_4c_4)u_4 + \\ & (z_5s_{45} + z_6c_{45})u_5 + (z_7s_6 + z_8c_6)u_6 + (z_9s_{67} + z_{10}c_{67})u_7)/z_1\boldsymbol{b}_1 + \\ & (z_1u_2 + z_{11}u_3 - (z_3c_4 - z_4s_4)u_4 - \\ & (z_5c_{45} - z_6s_{45})u_5 - (z_7c_6 - z_8s_6)u_6 - (z_9c_{67} - z_{10}s_{67})u_7)/z_1\boldsymbol{b}_2 \end{aligned}$$

The acceleration of the center of mass is given by:

$${}^A\boldsymbol{a}^{\mathrm{CM}} = \frac{1}{z_1}\left[\left\{\sum_{i=1}^{8} M(1,i)\dot{u}(i) - f_I(1)\right\}\boldsymbol{b}_1 + \left\{\sum_{i=1}^{8} M(2,i)\dot{u}(i) - f_I(2)\right\}\boldsymbol{b}_2\right]$$

Once the equations of motions are derived, the kinetic energy is easily calculated as

$$\text{Kinetic Energy} = \frac{1}{2}\sum_{i=1}^{8}\sum_{j=1}^{8} u(i)M(i,j)u(j)$$

~